

to the describing and claiming of the invention of the subject case in a U.S. patent application. Based on this information and pursuant to 37 CFR 1.56(b), please prepare and file the proper Information Disclosure Statement or equivalent document.

INVENTOR'S INFORMATION

☆ PATENT NUMBER, PUBLICATION ; INVENTOR(S), AUTHOR(S) ; DATE etc.

Japanese Patent Disclosure (Kokai) No. 10-177624 ; H.Sasaki et al ; June 30, 1998

★ CONCISE EXPLANATION

This discloses a method of word matching. The distance between a word and a template are defined as the sum of distances between the characters and the templates.

☆ Japanese Patent Disclosure (Kokai) No. 08-167008 ; A.Suzuki et al ; June 25, 1996

★ This discloses a method of word matching. ~~It define~~ The confidence level of a recognized word is defined as the number of successfully recognized characters.

☆ ~~The proceedings of Meeting on Image Recognition and Understanding 2000 (MIRU 2000)~~  
~~Vol II, Pt 6 ; T.Hamamura et al ; July, 2000~~

★ ~~Inventor wrote this paper.~~

PRIOR APPLICATION(S) OF INVENTOR(S) OR OF KABUSHIKI KAISHA TOSHIBA (ASSIGNEE)  
APPLICATION NUMBER TOSHIBA REFERENCE COUNTRY

AGENT

MEMO

INVENTOR(S)  
SIGNATURE & DATE

CHECKED BY  
11 Aug. 2000  
Bunpei Irie

Tomoyuki Hamamura August 11, 2000

PATENT ENGINEER'S INFORMATION

PATENT ENGINEER'S COMMENT ON INVENTOR(S) INFORMATION OR PATENT ENGINEER'S INFORMATION

☆

None.

CHECKED BY PATENT ENGINEER(S)  
SIGNATURE & DATE

11 Aug. 2000  
Akira Toshima

Akira Toshima Aug. 16, 2000

(to U.S.  
Attorney)

000751

TOSHIBA REFERENCE

506 31059

JAPANESE AGENT REFERENCE

Sheet

of

(11)特許出願公開番号

(43)公開日 平成10年(1998)6月30日

**620B**



## 【特許請求の範囲】

【請求項1】 文字認識対象の入力文字列ボタンから連続した複数の候補文字ボタンを、その位置情報を座標メモリに記録することにより抽出すると共に、該候補文字ボタンの文字認識を行い、該記録された位置情報を利用して複数の候補文字コード列を作成し、前記候補文字ボタンの認識結果に基づき前記候補文字コード列の中から最良文字コード列を選択して前記入力文字列ボタンの文字切出し位置を決定する文字切出し方法において、前記文字認識を、前記候補文字ボタンに対応した文字コードであって、類似度の高い順に並べられた上位の一定個数の当該文字コードを、前記認識結果として、取得するステップとしたとき、

(a) 前記文字コードを前記候補文字ボタンごとに文字コード格納部に記録するステップと、

(b) 前記文字コード格納部に記録した文字コードにその文字種に応じた文字評価値を付与し、該文字評価値を前記文字コードに対応した前記文字コード格納部の格納場所に格納するステップと、

(c) 前記座標メモリに記録した位置情報と前記文字コード格納部に記録した文字評価値とを参照して、前記候補文字コード列を作成するステップとを含むことを特徴とする文字切出し方法。

【請求項2】 請求項1に記載の文字切出し方法において、前記候補文字ボタンをセグメントおよび新規セグメントとしたとき、

前記セグメントを前記入力文字列ボタンの各黒ブロック領域として抽出して、該黒ブロック領域の位置座標を前記座標メモリに記録するステップと、

前記新規セグメントを、前記抽出したセグメント同士を各々の位置座標に基づいて統合することにより生成して、該新規セグメントの位置座標を前記座標メモリに追加して記録するステップとを以て前記候補文字ボタンの抽出を行うことを特徴とする文字切出し方法。

【請求項3】 請求項2に記載の文字切出し方法において、

前記新規セグメントの生成は、

前記記録したセグメント $S_n$  ( $n$ は整数)の位置情報を、前記座標メモリから読み出すステップと、

前記入力文字列ボタン中の前記セグメント $S_n$ に関して一定の側にあるセグメント $S_k$  ( $k$ は整数)の位置情報を、前記座標メモリから読み出すステップと、

セグメント $S_n$ とセグメント $S_k$ との間の距離 $D_{nk}$ を前記読み出した各々の位置情報から求めるステップと、

前記入力文字列ボタンの行高さ $L$ の定数 $e$  ( $e$ は正の実数)倍と前記求めた距離 $D_{nk}$ とを比較するステップと、該比較結果が $D_{nk} \leq e \cdot L$ のときにセグメント $S_n$ とセグメント $S_k$ とを統合して新規セグメントを生成するステップとを以て行うことを特徴とする文字切出し方法。

【請求項4】 請求項3に記載の文字切出し方法において、前記定数 $e$ の値を1.2に設定したことを特徴とする文字切出し方法。

【請求項5】 請求項1に記載の文字切出し方法において、

前記文字評価値を、前記文字コードの類似度と予め設定した文字種に応じた制約係数とを積算することにより求めることを特徴とする文字切出し方法。

【請求項6】 請求項5に記載の文字切出し方法において、

前記文字種に応じた制約係数を、該文字種が前記入力文字列ボタン中に出現する確率の値としたことを特徴とする文字切出し方法。

【請求項7】 請求項1に記載の文字切出し方法において、

前記文字評価値を文字種に応じた制約係数とし、該制約係数は、該文字種が前記入力文字列ボタン中出现するか否かを「1」または「0」のそれぞれ2値で表した値であることを特徴とする文字切出し方法。

【請求項8】 請求項2に記載の文字切出し方法において、

前記(c)ステップは、

前記記録した候補文字ボタンの位置情報を前記座標メモリから読み出して、該位置情報に基づき各候補文字ボタンの始点位置および終点位置を切出し候補位置 $C_i$  ( $i$ は整数)として求め、これら候補文字ボタンと切出し候補位置との対応関係をテーブルメモリ部に記録するステップと、

前記記録した対応関係を参照して作成関数 $F(C_i, P)$ を用いた処理を行い、候補文字番号列を作成するステップと、

前記作成した候補文字番号列と前記文字コード格納部に記録した文字評価値とを参照して候補文字コード列を作成するステップとを含むことを特徴とする文字切出し方法。但し、切出し候補位置 $C_i$ を、入力文字列ボタン方向に順次に整理するように番号付けしており、始点位置としての切出し候補位置 $C_i$ およびパス $P$ を引き数とする作成関数 $F(C_i, P)$ は、

(e1) 前記テーブルメモリ部に記録した切出し候補位置を始点位置 $C_i$ として指定して、該始点位置を指定順にメモリ手段に記録しておく処理と、

(e2) 前記指定した始点位置 $C_i$ が前記入力文字列ボタンの最終端位置か否かを判別する処理と、

(e3) 前記始点位置 $C_i$ が最終端位置でない場合には、

前記始点位置 $C_i$ に対して指定が可能な前記記録した切出し候補位置を終点位置 $C_j$  ( $j$ は整数)として指定するステップと、

前記対応関係から前記指定した始点位置 $C_i$ および終点位置 $C_j$ により候補文字番号を指定して、該候補文字番

号をパスPに格納するステップと、  
作成関数  $F(C_i, P)$  の処理を開始するステップとによる処理と、

(e4) 前記始点位置  $C_i$  が最終端位置である場合には、  
前記パスPを文字番号格納部に保存するステップと、  
全候補文字番号が前記文字番号格納部に記録されたか否かを判別するステップと、  
全候補文字番号が前記文字番号格納部に記録された場合には処理を終了するステップと、  
全候補文字番号が前記文字番号格納部に記録されていない場合には、前記メモリ手段を参照して前記始点位置  $C_i$  より2つ前に記録されている切出し候補位置  $C_r$  と最終端位置との間の候補文字番号を前記パスPから消去し、前記メモリ手段から、前記切出し候補位置  $C_r$  と、最終端位置と、前記切出し候補位置  $C_r$  および最終端位置間の切出し候補位置とを消去するステップと、  
作成関数  $F(C_r, P)$  の処理を開始するステップとによる処理とを実行する関数である。

【請求項9】 請求項6に記載の文字切出し方法において、  
前記文字コード格納部から読み出した前記文字評価値を前記作成した候補文字コード列に従い加算し、該加算の結果により類似度が最大となる候補文字コード列を前記最良文字コード列として選択することを特徴とする文字切出し方法。

【請求項10】 請求項7に記載の文字切出し方法において、  
前記文字コード格納部を参照して前記作成した候補文字コード列と単語辞書との照合を行い該当する単語の有無を調べ、  
① 該当した単語が複数あった場合には、それら該当単語を構成する文字コードの前記類似度の順位の合計が最も小さくなる候補文字コード列を前記最良文字コード列として選択し、  
② 該当した単語が無かった場合には、前記候補文字コード列を構成する文字コードの前記類似度の順位の合計が最も小さくなる候補文字コード列を前記最良文字コード列として選択することを特徴とする文字切出し方法。

【請求項11】 文字認識対象の入力文字列ボタンを含む原画像を読み取り、該読み取った原画像を格納する画像メモリを具えた画像入力部と、文字認識対象の入力文字列ボタンから連続した複数個の候補文字ボタンを抽出してその位置情報を座標メモリに記録する候補文字ボタン抽出部と、前記候補文字ボタンの文字認識を行う文字認識部と、前記記録された位置情報を利用して複数の候補文字コード列を作成する候補文字コード列作成部と、前記候補文字ボタンの認識結果に基づき前記候補文字コード列の中から最良文字コード列を選択して前記入力文

字列ボタンの文字切出し位置を決定する最良文字コード列選択部とを具える文字切出し装置において、  
前記文字認識部を、前記候補文字ボタンに対応した文字コードであって、類似度の高い順に並べられた上位の一定個数の当該文字コードを、前記認識結果として、前記候補文字ボタンごとに文字コード格納部に記録する手段とし、

前記文字コード格納部に記録されている文字コードにその文字種に応じた文字評価値を付与して、該文字評価値を前記文字コードに対応した前記文字コード格納部の格納場所に格納する文字評価部を具え、  
前記候補文字コード列作成部を、前記座標メモリに記録されている位置情報と、前記文字コード格納部に記録されている文字評価値とを参照して、前記候補文字コード列を作成する手段としたことを特徴とする文字切出し装置。

【請求項12】 請求項11に記載の文字切出し装置において、  
前記候補文字ボタン抽出部は、  
前記入力文字列ボタンの各黒ブロック領域をセグメントとして抽出し、該セグメントの位置座標を前記座標メモリに記録するセグメント抽出部と、  
前記抽出されたセグメント同士を各々の位置座標に基づいて統合することにより新規セグメントを生成し、該新規セグメントの位置座標を前記座標メモリに追加して記録するセグメント統合部とを具えており、  
前記セグメントの位置座標と前記新規セグメントの位置座標とを前記座標メモリに記録することにより、該記録された位置座標を前記候補文字ボタンの位置座標として、前記候補文字ボタンの抽出を行うことを特徴とする文字切出し装置。

【請求項13】 請求項12に記載の文字切出し装置において、  
前記セグメント統合部は、  
前記記録されたセグメント  $S_n$  ( $n$ は整数)の位置情報を、前記座標メモリから読み出す第1読出部と、  
前記入力文字列ボタン中の前記セグメント  $S_n$  に関して一定の側にあるセグメント  $S_k$  ( $k$ は整数)の位置情報を、前記座標メモリから読み出す第2読出部と、  
セグメント  $S_n$  とセグメント  $S_k$  との間の距離  $D_{nk}$  を前記読み出された各々の位置情報から求める距離検出部と、  
前記入力文字列ボタンの行高さ  $L$  の定数  $e$  ( $e$ は正の実数)倍と前記求められた距離  $D_{nk}$  とを比較する比較部と、  
該比較結果が  $D_{nk} \leq e \cdot L$  のときにセグメント  $S_n$  とセグメント  $S_k$  とを統合して新規セグメントを生成するセグメント生成部と、  
前記生成された新規セグメントの位置情報を前記座標メモリに追加して記録する書込部とを具えることを特徴と

する文字切出し装置。

【請求項14】 請求項13に記載の文字切出し装置において、前記定数 $e$ の値が1、2に設定されていることを特徴とする文字切出し装置。

【請求項15】 請求項11に記載の文字切出し装置において、前記文字評価部は、前記文字評価値を、前記文字コードの類似度と予め設定した文字種に応じた制約係数との積として求めるための評価値計算部を具えることを特徴とする文字切出し装置。

【請求項16】 請求項15に記載の文字切出し装置において、前記文字種に応じた制約係数を、該文字種が前記入力文字列ボタン中に出現する確率の値としたことを特徴とする文字切出し装置。

【請求項17】 請求項11に記載の文字切出し装置において、前記文字評価部は、前記文字評価値として文字種に応じた制約係数を用いて処理を行い、該制約係数を、該文字種が前記入力文字列ボタンに出現するか否かを「1」または「0」のそれぞれ2値で表した値としたことを特徴とする文字切出し装置。

【請求項18】 請求項12に記載の文字切出し装置において、前記候補文字コード列作成部は、前記記録された候補文字ボタンの位置情報を前記座標メモリから読み出して、該位置情報に基づき各候補文字ボタンの始点位置および終点位置を切出し候補位置 $C_i$ （ $i$ は整数）として求め、これら候補文字ボタンと切出し候補位置との対応関係をテーブルメモリ部に記録するテーブル作成部と、前記記録された対応関係を参照して作成関数 $F(C_i, P)$ を用いた処理を行うことにより候補文字番号列を作成し、該候補文字番号列と前記文字コード格納部に記録された文字評価値とを参照して候補文字コード列を作成する処理回路と、前記候補文字番号の配列情報を記録するための候補文字記録部と、前記配列情報を前記候補文字列として保存するための文字列格納部と、前記作成された候補文字コード列を格納するための文字コード列格納部とを具えていることを特徴とする文字切出し装置。但し、切出し候補位置 $C_i$ を、入力文字列ボタン方向に順次に整列するように番号付けしてあり、始点位置としての切出し候補位置 $C_i$ およびパス $P$ を引き数とする作成関数 $F(C_i, P)$ は、

（e1）前記テーブルメモリ部に記録した切出し候補位置を始点位置 $C_i$ として指定して、該始点位置を指定順にメモリ手段に記録しておく処理と、

（e2）前記指定した始点位置 $C_i$ が前記入力文字列パ

タンの最終端位置か否かを判別する処理と、

（e3）前記始点位置 $C_i$ が最終端位置でない場合には、前記始点位置 $C_i$ に対して指定が可能な前記記録した切出し候補位置を終点位置 $C_j$ （ $j$ は整数）として指定するステップと、前記対応関係から前記指定した始点位置 $C_i$ および終点位置 $C_j$ により候補文字番号を指定して、該候補文字番号をパス $P$ に格納するステップと、作成関数 $F(C_j, P)$ の処理を開始するステップとによる処理と、

（e4）前記始点位置 $C_i$ が最終端位置である場合には、前記パス $P$ を文字列格納部に保存するステップと、全候補文字番号が前記文字番号格納部に記録されたか否かを判別するステップと、全候補文字番号が前記文字番号格納部に記録された場合には処理を終了するステップと、全候補文字番号が前記文字番号格納部に記録されていない場合には、前記メモリ手段を参照して前記始点位置 $C_i$ より2つ前に記録されている切出し候補位置 $C_i$ と最終端位置との間の候補文字番号を前記パス $P$ から消去し、前記メモリ手段から、前記切出し候補位置 $C_i$ と、最終端位置と、前記切出し候補位置 $C_i$ および最終端位置間の切出し候補位置とを消去するステップと、作成関数 $F(C_i, P)$ の処理を開始するステップとによる処理とを実行する関数である。

【請求項19】 請求項16に記載の文字切出し装置において、前記最良文字コード列選択部は、前記文字コード格納部から読み出された前記文字評価値を前記作成した候補文字コード列に従い加算し、該加算の結果により類似度が最大となる候補文字コード列を前記最良文字コード列として選択することを特徴とする文字切出し装置。

【請求項20】 請求項17に記載の文字切出し装置において、前記最良文字コード列選択部は、前記文字コード格納部を参照して前記候補文字コード列作成部で作成された前記候補文字コード列と単語辞書との照合を行い該当する単語の有無を調べ、

① 該当した単語が複数あった場合には、それら該当単語を構成する文字コードの前記類似度の順位の合計が最も小さくなる候補文字コード列を前記最良文字コード列として選択し、

② 該当した単語が無かった場合には、前記候補文字コード列を構成する文字コードの前記類似度の順位の合計が最も小さくなる候補文字コード列を前記最良文字コード列として選択することを特徴とする文字切出し装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、手書き文字の認識を行うに当たって入力文字列ボタンから文字の切出しを行う装置および方法に関する。

【0002】

【従来の技術】手書き文字ボタンは、文字ボタン間隔や文字ボタン形状の標準文字ボタンとの相違の度合いが大きいため、「一定間隔で文字を切り出す」といった従来手法では十分な文字切出し精度が得られない。この問題に対して、文献1「電子通信学会論文誌'86/9 Vol. J69-D No. 9 pp1292-1301」や文献2「特開平3-225579」に開示されているように、候補文字ラティスの手法を用いた文字切出し方法が提案されている。この文献1に開示されている方法によれば、入力文字列ボタンから要素矩形を抽出し、隣接する要素矩形同士の網羅的な組合せ（以下、候補文字と称する。）を生成してそれらの認識処理を行い、全候補文字の認識結果を評価して最適な文字ボタン（文字列）を選択することにより、文字間の接続関係を取り入れた入力文字列ボタンの文字切出しを行う。また、文献1に開示の方法によれば、単に認識結果の評価が最高となる候補文字の組合せが文字列として選択されてしまうが、文献2に開示の方法によれば、単語知識を導入することにより、単語として文字間の接続性が考慮された文字列が選択できる。

【0003】ところで、上述した文献1および文献2に開示されている従来手法によれば、全候補文字ボタンの認識処理と文字ボタンの評価とを行わなければならないから、入力文字列ボタンが長くなると探索空間が非常に大きくなり、このため膨大な演算量を必要とする。そこで、文献3「特開平6-195508」に開示されているように、最初に入力文字列から形状的知識に基づき文字切出しを行い、次に単語の接続関係を利用して不当な文字切出し部分を抽出し、その部分だけを要素矩形に分離して網羅的な文字切出しを行うことにより、探索空間の削減を図っている。

【0004】

【発明が解決しようとする課題】しかしながら、上述した文献3に開示の手法も網羅的な文字切出しを行う手法に変わらないため、探索空間の削減が不十分であり、このため演算量の増大といった問題を解決できていない。

【0005】従って、従来より、従来方法に比べて探索空間の削減が可能である文字切出し方法の出現と、この文字切出し方法に従って文字切出し処理を行う文字切出し装置の出現とが望まれていた。

【0006】

【課題を解決するための手段】そこで、この発明の文字切出し方法によれば、文字認識対象の入力文字列ボタンから連続した複数の候補文字ボタンを、その位置情報を座標メモリに記録することにより抽出すると共に、これら候補文字ボタンの文字認識を行い、この記録された位置情報を利用して複数の候補文字コード列を作成し、

前記候補文字ボタンの認識結果に基づき前記候補文字コード列の中から最良文字コード列を選択して前記入力文字列ボタンの文字切出し位置を決定する文字切出し方法において、前記文字認識を、前記候補文字ボタンに対応した文字コードであって、類似度の高い順に並べられた上位の一定個数の当該文字コードを、前記認識結果として、取得するステップとしたとき、（a）前記文字コードを前記候補文字ボタンごとに文字コード格納部に記録するステップと、（b）前記文字コード格納部に記録した文字コードにその文字種に応じた文字評価値を付与し、この文字評価値を前記文字コードに対応した前記文字コード格納部の格納場所に格納するステップと、

（c）前記座標メモリに記録した位置情報と前記文字コード格納部に記録した文字評価値とを参照して、前記候補文字コード列を作成するステップとを含むことを特徴とする。

【0007】このように、この発明の文字切出し方法は、各候補文字ボタンの文字コードを類似度に基づいて取得し、取得した文字コードにその文字種に応じた文字評価値を付与し、その文字評価値を参照して候補文字コード列を作成するといった各ステップを含んでいる。そして、上述の文字評価値を、例えば、取得した文字コードの文字種が、1）入力文字列ボタンに出現する文字種であるか、2）入力文字列ボタンに出現しない文字種であるかに基づく値とすることにより、この2）に相当する文字評価値が付与された文字コードを、候補文字コード列を作成する段階で除外することができる。よって、探索空間の削減が図れる。但し、文字評価値としては、上述の1）または2）の2値である必要はなく、1）であるか2）であるかを確率的に表現した値を文字評価値として用いても良い。この場合には、付与した文字評価値を適当な閾値と比較して、候補文字コード列を構成する文字コードであるか、候補文字コード列を構成しない文字コードであるかを決めれば良い。

【0008】また、この発明の文字切出し方法の好適な実施例によれば、前記候補文字ボタンをセグメントまたは新規セグメントの各々としたとき、前記セグメントを前記入力文字列の各黒ブロック領域として抽出して、これら黒ブロック領域の位置座標を前記座標メモリに記録するステップと、前記新規セグメントを、前記抽出したセグメント同士を各々の位置座標に基づいて統合することにより生成して、この新規セグメントの位置座標を前記座標メモリに追加して記録するステップとを以て前記候補文字ボタンの抽出を行うことを特徴とする。

【0009】このように、セグメントを抽出し、抽出した各セグメントの統合を行うことにより新規セグメントを生成して、上述の候補文字ボタンを抽出することができる。

【0010】この発明の文字切出し方法において、好ましくは、前記新規セグメントの生成は、前記記録したセ

グメント $S_n$  ( $n$ は整数)の位置情報を、前記座標メモリから読み出すステップと、前記入力文字列ボタン中の前記セグメント $S_n$ に関して一定の側にあるセグメント $S_k$  ( $k$ は整数)の位置情報を、前記座標メモリから読み出すステップと、セグメント $S_n$ とセグメント $S_k$ との間の距離 $D_{nk}$ を前記読み出した各々の位置情報から求めるステップと、前記入力文字列ボタンの行高さ $L$ の定数 $e$  ( $e$ は正の実数)倍と前記求めた距離 $D_{nk}$ とを比較するステップと、この比較結果が $D_{nk} \leq e \cdot L$ のときにセグメント $S_n$ とセグメント $S_k$ とを統合して新規セグメントを生成するステップとを以て行うのが良い。

【0011】このように、隣接するセグメント間の距離に基づいて、この距離と行高さとを比較することにより、セグメントの組を統合するか否かを判定することができる。ここで、上述の距離は画像上あるいは情報媒体上におけるセグメント間の距離に比例した量であり、例えば、各セグメントの始端位置同士を結ぶ入力文字列ボタン方向に平行な直線の長さを用いることができる。また、上述の行高さには、入力文字列ボタン方向に垂直な方向の黒ブロック領域の長さの最大値を用いることができる。

【0012】また、この発明の文字切出し方法において、好ましくは、前記定数 $e$ の値を1.2に設定するのが良い。この値は、繰返しテストを行って経験的に定めた値であり、読出し自在にメモリ手段に記憶させてある。

【0013】また、この発明の文字切出し方法の好適な実施例によれば、前記(c)ステップは、前記記録した候補文字ボタンの位置情報を前記座標メモリから読み出して、この位置情報に基づき各候補文字ボタンの始点位置および終点位置を切出し候補位置 $C_i$  ( $i$ は整数)として求め、これら候補文字ボタンと切出し候補位置との対応関係をテーブルメモリ部に記録するステップと、前記記録した対応関係を参照して作成関数 $F(C_i, P)$ を用いた処理を行い、候補文字番号列を作成するステップと、前記作成した候補文字番号列と前記文字コード格納部に記録した文字評価値とを参照して候補文字コード列を作成するステップとを含むことを特徴とする。但し、切出し候補位置 $C_i$ を、入力文字列ボタン方向に順次に整列するように番号付けしてあり、始点位置としての切出し候補位置 $C_i$ およびパス $P$ を引き数とする作成関数 $F(C_i, P)$ は、(e1)前記テーブルメモリ部に記録した切出し候補位置を始点位置 $C_i$ として指定して、該始点位置を指定順にメモリ手段に記録しておく処理と、(e2)前記指定した始点位置 $C_i$ が前記入力文字列ボタンの最終端位置か否かを判別する処理と、(e3)前記始点位置 $C_i$ が最終端位置でない場合には、前記始点位置 $C_i$ に対して指定が可能な前記記録した切出し候補位置を終点位置 $C_j$  ( $j$ は整数)として指定するステップと、前記対応関係から前記指定した始点位置 $C$

$i$ および終点位置 $C_j$ により候補文字番号を指定して、該候補文字番号をパス $P$ に格納するステップと、作成関数 $F(C_j, P)$ の処理を開始するステップとによる処理と、(e4)前記始点位置 $C_i$ が最終端位置である場合には、前記パス $P$ を文字番号格納部に保存するステップと、全候補文字番号が前記文字番号格納部に記録されたか否かを判別するステップと、全候補文字番号が前記文字番号格納部に記録された場合には処理を終了するステップと、全候補文字番号が前記文字番号格納部に記録されていない場合には、前記メモリ手段を参照して前記始点位置 $C_i$ より2つ前に記録されている切出し候補位置 $C_i$ と最終端位置との間の候補文字番号を前記パス $P$ から消去し、前記メモリ手段から、前記切出し候補位置 $C_i$ と、最終端位置と、前記切出し候補位置 $C_i$ および最終端位置間の切出し候補位置とを消去するステップと、作成関数 $F(C_i, P)$ の処理を開始するステップとによる処理とを実行する関数である。

【0014】このように、ある候補文字番号から別々の全ての候補文字番号を、切出し候補位置を指定することにより辿り、その候補文字番号を配列情報としてグループ化する上述の方法によれば、この配列情報のそれぞれを各候補文字番号列として得ることができる。そして、この候補文字番号列と文字コード格納部に記録されている文字評価値とを参照して、文字コード格納部に格納されている文字コードを読み出して配列させ、候補文字コード列を作成する。この候補文字コード列の作成にあつては、前述した文字評価値に基づいて対象としない文字コードに制約を課しているもので、この文字コードを除外することができ、よって、探索空間の削減が図れる。

【0015】また、この発明の文字切出し方法の好適な実施例によれば、前記文字評価値を、前記文字コードの類似度と予め設定した文字種に応じた制約係数とを積算することにより求めることを特徴とする。

【0016】このようにすると、類似度を取り入れた文字評価値を用いることにより、候補文字コード列の作成時に、類似度の低い文字コードを除外することができ、従って、探索空間の削減が図れる。また、制約係数には、例えば、入力文字列ボタンに出現する文字種の文字コードを候補文字コード列の構成対象とし、入力文字列ボタンに出現しない文字種の文字コードを候補文字コード列の構成対象としない、といった情報を含ませた値を設定することにより、候補文字コード列を作成する際に対象とする文字コードの個数を削減することができ、よって、探索空間の削減が図れる。

【0017】尚、この場合において、好ましくは、前記文字種に応じた制約係数を、この文字種が前記入力文字列ボタン中に出現する確率の値とするのが良い。

【0018】さらに、この発明の文字切出し方法において、好ましくは、前記文字コード格納部から読み出した前記文字評価値を前記作成した候補文字コード列に従い



加算し、この加算の結果により類似度が最大となる候補文字コード列を前記最良文字列として選択するのが良い。

【0019】また、この発明の文字切出し方法の好適な実施例によれば、前記文字評価値を文字種に応じた制約係数とし、この制約係数は、この文字種が前記入力文字列ボタンに出現するか否かを「1」または「0」のそれぞれ2値で表した値であることを特徴とする。

【0020】このように制約係数を設定すれば、「0」の制約係数が設定された文字コードを候補文字コード列の作成対象とせず、「1」の制約係数が設定された文字コードだけを候補文字コード列の作成対象とすることができるので、探索空間の削減が図れる。

【0021】また、この場合において、好ましくは、前記文字コード格納部を参照して前記作成した候補文字コード列と単語辞書との照合を行い該当する単語の有無を調べ、① 該当した単語が複数あった場合には、それら該当単語を構成する文字コードの前記類似度の順位の合計が最も小さくなる候補文字コード列を前記最良文字コード列として選択し、② 該当した単語が無かった場合には、前記候補文字コード列を構成する文字コードの前記類似度の順位の合計が最も小さくなる候補文字コード列を前記最良文字コード列として選択するのが良い。

【0022】次に、この発明の文字切出し装置によれば、文字認識対象の入力文字列ボタンを含む原画像を読み取り、この読み取った原画像を格納する画像メモリを具えた画像入力部と、文字認識対象の入力文字列ボタンから連続した複数の候補文字ボタンを抽出してその位置情報を座標メモリに記録する候補文字ボタン抽出部と、前記候補文字ボタンの文字認識を行う文字認識部と、前記記録された位置情報を利用して複数の候補文字コード列を作成する候補文字コード列作成部と、前記候補文字ボタンの認識結果に基づき前記候補文字コード列の中から最良文字コード列を選択して前記入力文字列ボタンの文字切出し位置を決定する最良文字コード列選択部とを具える文字切出し装置において、前記文字認識部を、前記候補文字ボタンに対応した文字コードであって、類似度の高い順に並べられた上位の一定個数の当該文字コードを、前記認識結果として、前記候補文字ボタンごとに文字コード格納部に記録する手段とし、前記文字コード格納部に記録されている文字コードにその文字種に応じた文字評価値を付与して、この文字評価値を前記文字コードに対応した前記文字コード格納部の格納場所に格納する文字評価部を具え、前記候補文字コード列作成部を、前記座標メモリに記録されている位置情報と、前記文字コード格納部に記録されている文字評価値とを参照して、前記候補文字コード列を作成する手段としたことを特徴とする。

【0023】このように、この発明の文字切出し装置は、各候補文字ボタンの文字コードを類似度に基づいて

取得する文字認識部と、取得した文字コードにその文字種に応じた文字評価値を付与する文字評価部と、その文字評価値を参照して候補文字コード列を作成する候補文字コード列作成部とを具えている。そして、上述の文字評価値を、例えば、文字認識部で取得した文字コードの文字種が、1) 入力文字列に出現する文字種であるか、2) 入力文字列に出現しない文字種であるかに基づく値とすることにより、この2)に相当する文字評価値が付与された文字コードを、候補文字コード列を作成する段階で除外することができる。よって、探索空間の削減が図れる。

【0024】但し、文字評価値としては、上述の1)または2)の2値である必要はなく、1)であるか2)であるかを確率的に表現した値を文字評価値として用いても良い。そして、この文字評価値を適当な閾値と比較することにより、候補文字コード列を構成する文字コードであるか、候補文字コード列を構成しない文字コードであるかを決定する構成としても良い。

【0025】また、この発明の文字切出し装置の好適な構成例によれば、前記候補文字ボタン抽出部は、前記入力文字列ボタンの各黒ブロック領域をセグメントとして抽出し、これらセグメントの位置座標を前記座標メモリに記録するセグメント抽出部と、前記抽出されたセグメント同士を各々の位置座標に基づいて統合することにより新規セグメントを生成し、この新規セグメントの位置座標を前記座標メモリに追加して記録するセグメント統合部とを具えており、前記セグメントの位置座標と前記新規セグメントの位置座標とを前記座標メモリに記録することにより、この記録された位置座標を前記候補文字ボタンの位置座標として、前記候補文字ボタンの抽出を行うことを特徴とする。

【0026】このように、セグメントを抽出し、抽出した各セグメントの統合を行うことにより新規セグメントを生成して、上述の候補文字ボタンを抽出することができる。

【0027】また、この発明の文字切出し装置において、好ましくは、前記セグメント統合部は、前記記録されたセグメント $S_n$  ( $n$ は整数)の位置情報を、前記座標メモリから読み出す第1読出部と、前記入力文字列ボタン中の前記セグメント $S_n$ に関して一定の側にあるセグメント $S_k$  ( $k$ は整数)の位置情報を、前記座標メモリから読み出す第2読出部と、セグメント $S_n$ とセグメント $S_k$ との間の距離 $D_{nk}$ を前記読み出された各々の位置情報から求める距離検出部と、前記入力文字列ボタンの行高さ $L$ の定数 $e$  ( $e$ は正の実数)倍と前記求められた距離 $D_{nk}$ とを比較する比較部と、この比較結果が $D_{nk} \leq e \cdot L$ のときにセグメント $S_n$ とセグメント $S_k$ とを統合して新規セグメントを生成するセグメント生成部と、前記生成された新規セグメントの位置情報を前記座標メモリに追加して記録する書込部とを具えるのが良



い。

【0028】このように構成すると、セグメント統合部は、隣接するセグメント間の距離に基づいて、この距離と行高さとを比較することにより、セグメントの組を統合するか否かを判定することができる。

【0029】また、この発明の文字切出し装置において、好ましくは、前記定数 $e$ の値が1、2に設定されているのが良い。この値は、繰返しテストを行って経験的に定めた値であり、読出し自在にメモリ手段に記憶させてある。

【0030】また、この発明の文字切出し装置の好適な構成例によれば、前記候補文字コード列作成部は、前記記録した候補文字パタンの位置情報を前記座標メモリから読み出して、該位置情報に基づき各候補文字パタンの始点位置および終点位置を切出し候補位置 $C_i$  ( $i$ は整数)として求め、これら候補文字ボタンと切出し候補位置との対応関係をテーブルメモリ部に記録するテーブル作成部と、前記記録された対応関係を参照して作成関数 $F(C_i, P)$ を用いた処理を行うことにより候補文字番号列を作成し、この候補文字番号列と前記文字コード格納部に記録された文字評価値とを参照して候補文字コード列を作成する処理回路と、前記候補文字番号の配列情報列を記録するための候補文字記録部と、前記配列情報を前記候補文字番号列として保存するための文字列格納部と、前記作成された候補文字コード列を格納するための文字コード列格納部とを具備していることを特徴とする。但し、切出し候補位置 $C_i$ を、入力文字列方向に順次に整列するように番号付けしてあり、始点位置としての切出し候補位置 $C_i$ およびパス $P$ を引き数とする作成関数 $F(C_i, P)$ は、(e1)前記テーブルメモリ部に記録した切出し候補位置を始点位置 $C_i$ として指定して、該始点位置を指定順にメモリ手段に記録しておく処理と、(e2)前記指定した始点位置 $C_i$ が前記入力文字列パタンの最終端位置か否かを判別する処理と、(e3)前記始点位置 $C_i$ が最終端位置でない場合には、前記始点位置 $C_i$ に対して指定が可能な前記記録した切出し候補位置を終点位置 $C_j$  ( $j$ は整数)として指定するステップと、前記対応関係から前記指定した始点位置 $C_i$ および終点位置 $C_j$ により候補文字番号を指定して、該候補文字番号をパス $P$ に格納するステップと、作成関数 $F(C_j, P)$ の処理を開始するステップとによる処理と、(e4)前記始点位置 $C_i$ が最終端位置である場合には、前記パス $P$ を文字番号格納部に保存するステップと、全候補文字番号が前記文字番号格納部に記録されたか否かを判別するステップと、全候補文字番号が前記文字番号格納部に記録された場合には処理を終了するステップと、全候補文字番号が前記文字番号格納部に記録されていない場合には、前記メモリ手段を参照して前記始点位置 $C_i$ より2つ前に記録されている切出し候補位置 $C_i$ と最終端位置との間の候補文字番号を前記パス $P$

から消去し、前記メモリ手段から、前記切出し候補位置 $C_i$ と、最終端位置と、前記切出し候補位置 $C_i$ および最終端位置間の切出し候補位置とを消去するステップと、作成関数 $F(C_i, P)$ の処理を開始するステップとによる処理とを実行する関数である。

【0031】このように、上述の構成の候補文字コード列作成部は、ある候補文字番号から別々の全ての候補文字番号を、切出し候補位置を指定することにより辿り、その候補文字番号を配列情報としてグループ化することにより、この配列情報のそれぞれを候補文字番号列として取得することができる。また、上述の処理回路は、この候補文字番号列と文字コード格納部に記録されている文字評価値とを参照して、文字コード格納部に格納されている文字コードを読み出して配列させ、候補文字コード列を作成する。この候補文字コード列の作成にあつては、前述した文字評価値に基づいて対象としない文字コードに制約を課すため、この文字コードを除外することができ、よって、探索空間の削減が図れる。

【0032】また、この発明の文字切出し装置の好適な構成例によれば、前記文字評価部は、前記文字評価値を、前記文字コードの類似度と予め設定した文字種に応じた制約係数との積として求めるための評価値計算部を具備することを特徴とする。

【0033】このようにすると、文字評価部は、評価値計算部により類似度を取り入れた文字評価値を求めることができるので、候補文字コード列の作成時に、類似度の低い文字コードを除外することができ、従って、探索空間の削減が図れる。

【0034】また、制約係数には、例えば、入力文字列パタンに出現する文字種の文字コードを候補文字コード列の構成対象とし、入力文字列パタンに出現しない文字種の文字コードを候補文字コード列の構成対象としない、といった情報を含ませた値を設定することにより、候補文字コード列を作成する際に対象とする文字コードの個数を削減することができ、よって、探索空間の削減が図れる。

【0035】この場合において、好ましくは、前記文字種に応じた制約係数を、この文字種が前記入力文字列パタン中に出現する確率の値とするのが良い。

【0036】また、この発明の文字切出し装置において、好ましくは、前記最良文字コード列選択部は、前記文字コード格納部から読み出された前記文字評価値を前記作成した候補文字コード列に従い加算し、この加算の結果により類似度が最大となる候補文字コード列を前記最良文字コード列として選択するのが良い。

【0037】また、この発明の文字切出し装置の好適な構成例によれば、前記文字評価部は、前記文字評価値として文字種に応じた制約係数を用いて処理を行い、この制約係数を、この文字種が前記入力文字列パタンに出現するか否かを「1」または「0」のそれぞれ2値で表し

た値としたことを特徴とする。

【0038】このように制約係数を設定すれば、「0」の制約係数が設定された文字コードを候補文字コード列の作成対象としないで、「1」の制約係数が設定された文字コードだけを候補文字コード列の作成対象とすることができ、従って、探索空間の削減が図れる。

【0039】また、この発明の文字切出し装置において、好ましくは、前記最良文字コード列選択部は、前記文字コード格納部を参照して前記候補文字コード列作成部で作成された前記候補文字コード列と単語辞書との照合を行い該当する単語の有無を調べ、① 該当した単語が複数あった場合には、それら該当単語を構成する文字コードの前記類似度の順位の合計が最も小さくなる候補文字コード列を前記最良文字コード列として選択し、②

該当した単語が無かった場合には、前記候補文字コード列を構成する文字コードの前記類似度の順位の合計が最も小さくなる候補文字コード列を前記最良文字コード列として選択する構成とするのが良い。

【0040】

【発明の実施の形態】以下、図を参照して、この発明の実施の形態につき説明する。尚、図は、この発明の構成、配置関係および動作が理解できる程度に概略的に示しており、また、以下に記載する数値条件等は単なる好適例を示しているに過ぎない。従って、この発明は、この実施の形態に何ら限定されることがない。

【0041】【第1の実施の形態】図1は、第1の実施の形態の文字切出し装置の構成を示すブロック図である。図1に示すように、この実施の形態の文字切出し装置は、画像入力部10と、候補文字ボタン抽出部12と、候補文字コード列作成部14と、文字認識部16と、最良文字コード列選択部18とを具えている。また、この実施の形態の文字切出し装置は、位置情報を記録しておくための座標メモリ20と、文字コードを格納するための文字コード格納部22とを記憶手段として具えている。この実施の形態では、上述の画像入力部10、候補文字ボタン抽出部12、候補文字コード列作成部14および座標メモリ20の動作タイミングやデータ入出力のチェック等を制御部24により行っている。さらに、この実施の形態の文字切出し装置は、候補文字ボタンに対して求められた文字コードの評価を行うため、制約係数処理部26と、評価値計算部28とを具えており、この処理部26は計算部28の制御下にある。そして、この手段28と、上述した文字認識部16、文字コード格納部22および最良文字コード列選択部18とは、評価系制御部30により、その動作タイミングやデータ入出力等の管理が行われるように構成されている。この実施の形態では、上述の制約係数処理部26と、評価値計算部28と、評価系制御部30とを以て、文字評価部32と称している。そして、上述の制御部24と評価系制御部30との間は、各々の制御下にある各手段間

でデータの受け渡しが行えるように、結合されている。

【0042】尚、上述の最良文字コード列選択部18の出力は、例えばいわゆるコンピュータ装置に入力され、そこで文字認識情報として活用される。また、上述した候補文字ボタン抽出部12、候補文字コード列作成部14、文字認識部16、最良文字コード列選択部18、座標メモリ20、文字コード格納部22、制御部24、制約係数処理部26、評価値計算部28および評価系制御部30は、中央演算処理装置（CPU）、入出力部および記憶手段を具えたコンピュータ装置として、各要素のハードウェアを構成してもよいし、上述した各手段を、一つのコンピュータ装置にまとめた構成としてもよい。

【0043】また、図2は、この実施の形態の文字切出し装置の動作フローを示すフローチャートである。図2に示すように、この実施の形態の文字切出し装置による文字切出し処理は、制約係数入力（図2のS1）、画像入力（図2のS2）、セグメント抽出（図2のS3）、セグメント統合（図2のS4）、文字認識（図2のS5）、文字評価（図2のS6）、候補文字コード列作成（図2のS7）および最良文字コード列選択（図2のS8）のステップごとに順次に行われる。以下、この図2のフローに従い、文字切出し処理の手順について、各手段の構成および動作と共に説明する。

【0044】＜制約係数の入力＞始めに、上述の制約係数処理部26に制約係数を入力する（図2のS1）。この実施の形態で用いられる制約係数とは、或る文字種が入力文字列中に出現する確率を表す値のことである。この実施の形態では、数字・記号、カタカナ、平仮名および漢字の4種の文字種を想定しており、各文字種に対して制約係数を設定する。制約係数は、予め、文字認識対象である文書領域の各文字種ごとの出現頻度（分布）を求めることにより、その割合（比率）に応じた値として設定される。例えば、認識対象の帳票上に全く数字および記号が出現しない場合には、数字・記号の制約係数として0を設定する。逆に、対象とする帳票に、必ず漢字が出現する場合には、漢字の制約係数として1を設定する。また、対象とする帳票に平仮名が出現する頻度が50%であるときには、平仮名の制約係数として0.5を設定する。このように、各文字種ごとの制約係数を、対象とする文書領域に応じた0以上1以下の実数値として、制約係数処理部26に入力手段（例えば、キーボード）を用いて入力する。そして、制約係数処理部26は、入力された制約係数を、各文字種に対応した格納場所に、読出し自在に記憶させておく。

【0045】尚、上述したように、制約係数として0以上1以下の値を設定する必要は必ずしもない。例えば、各制約係数を、出現頻度に対応した0以上N以下の値（Nは整数）として設定してもよい。但し、この場合には、重み係数Wを、次式

$$W = \sqrt{(\alpha^2 + \beta^2 + \gamma^2 + \eta^2)}$$

により求めて、各制約係数を、 $\alpha/W$ 、 $\beta/W$ 、 $\gamma/W$  および  $\eta/W$  という具合に正規化しておく必要がある

(正規化を行わない場合、係数値が発散してしまうおそれがある。)。但し、記号 $\alpha$ 、 $\beta$ 、 $\gamma$ および $\eta$ はそれぞれ数字・記号、カタカナ、平仮名および漢字の制約係数を表す。

【0046】また、上述した制約係数は、文字認識対象となる文書領域の部分領域ごとに設定することもできる。例えば、表などが記載されている帳票を認識対象とする場合には、文字認識を行う対象領域が項目ごとに分かれている。そして、例えば、項目の「氏名」の欄を認識する場合、氏名として記号および数字の文字種が出現する頻度は統計的に皆無に等しいから、記号・数字の制約係数として0を設定する。従って、この領域の認識に限っては、各制約係数を上述の記号を用いて表せば、 $\alpha=0$ 、 $\beta=1$ 、 $\gamma=1$ および $\eta=1$ と設定するのが好適である。そして、他の項目が記載されている領域に対しては、その欄に記載されている内容に応じた制約係数を設定すればよい。このように、制約係数処理部26には、部分領域ごとに制約係数を設定することができる。この場合、制約係数処理部26には、制約係数を、文書領域中の位置座標に対応付けて、各文字種ごとに入力する。

【0047】この実施の形態の文字切出し処理の理解を容易にするため、図3に入力文字列パタンの一例を示す。図3には、入力文字列パタン34として、手書きの「弘三」という文字パタン(文字画像)が示してある。尚、この実施の形態では、入力文字列パタンが横書きである場合を想定しているが、これに限らず縦書きであっても構わない。以下、「弘三」という氏名が記載された欄を認識対象とする場合につき、上述した各ステップで行う処理を説明する。上述した通り、図3に示す入力文字列34に対しては、制約係数として、 $\alpha=0$ 、 $\beta=1$ 、 $\gamma=1$ および $\eta=1$ が制約係数処理部26に入力される。

【0048】<画像の入力>次に、画像入力のステップ(図2のS2)につき説明する。このステップでは、画像入力部10が、対象の文書領域を光学的に読み取って、読み取った画像データを記憶手段に記録する処理を行う。この処理は、制約係数の入力終了後に、オペレータが制御部24を介して、画像入力部10に指示を与えることにより開始される。

【0049】上述した画像入力部10は、文字認識対象の入力文字列パタンを含んだ原画像を読み取り、その読み取った原画像を格納するための画像メモリ36を具備している。また、従来公知の構成と同様に、図示せずとも帳票や原稿等の情報媒体を主走査方向および副走査方向に移動させる手段である走査機構(スキャナ)と、この情報媒体から原画像を光学的に読み取る光電変換部と、読み取った原画像から所定の領域を切り出すための切出し

部とを具備している。以上の構成により、画像入力部10は、情報媒体からの原画像の読出しと、この原画像の画像メモリ36への記録と、原画像からの所定領域(入力文字列)の切出しとを行う。尚、情報媒体からの原画像の読出しは、2値画像として読み取っても良いし、多値画像として読み取ってもよい。また、画像入力部10として、例えば、いわゆるタブレットを使用しても良い。

【0050】<候補文字パタンの抽出>次に、候補文字パタンを抽出するステップ(図2のS3およびS4)につき説明する。このステップの処理は、上述の候補文字パタン抽出部12により行われる。候補文字パタン抽出部12は、文字認識対象の入力文字列パタンから連続した複数個の候補文字パタンを抽出してその位置情報を座標メモリ20に記録する手段である。候補文字パタン抽出部12は、セグメント抽出部38と、セグメント統合部40とを具備している。そして、まず、セグメント抽出部38によりセグメントの抽出を行い(図2のS3)、続いて、セグメント統合部40によりセグメントの統合(新規セグメントの生成)を行う(図2のS4)。

【0051】これらのステップは、画像入力部10が画像の入力を終えたことを制御部24に伝達し、これに回答して制御部24がセグメント抽出部38に動作の開始を指示することにより開始される。あるいは、オペレータが、制御部24に、候補文字パタン抽出部12の動作を開始するように指示を与えてもよい。

【0052】上述のセグメント抽出部38は、画像入力部10で得られた入力文字列パタンの各黒ブロック領域をセグメントとして抽出し、このセグメントの位置座標を上述の座標メモリ20に記録するための手段である。

【0053】図3において、上述した入力文字列方向は、図中のX座標方向(主走査方向)である。また、図3において、上述した黒ブロック領域とは、各文字パタンに外接する矩形領域(セグメントと称している。)のことである。

【0054】図3の例では、図中の文字パタン「弓」(「弘」の偏)に外接するセグメント $S_0$ 、文字パタン「ム」(「弘」の旁)に外接するセグメント $S_1$ 、文字パタン「三」に外接するセグメント $S_2$ が、それぞれセグメント抽出部38により抽出される。これらセグメントは、入力文字列パタン34において、入力文字列方向に順次に $S_0$ 、 $S_1$ 、 $S_2$ という具合に配列している。

【0055】これらセグメントの抽出は、まず、入力文字列パタンのX方向に走査を行うことにより、入力文字列パタンをX方向に投影した射影分布すなわち黒点のヒストグラムを求める。そして、このヒストグラムの極小点をX軸上の切出し位置とする。次に、入力文字列パタンのX軸に垂直なY座標方向に走査を行うことにより、同様にヒストグラムを求める。そして、このヒストグラムの極小点をY軸上の切出し位置とする。このようにして、X軸上とY軸上の切出し位置で囲まれた矩形領

域が求められる。この実施の形態では、この矩形領域を求めることを、セグメントの抽出と称しており、上述した公知の方法で行える。

【0056】このように、セグメントの抽出は、求められた切出し位置同士の交点である4点（セグメントの各頂点に相当する。）の座標（画素位置）を検出することにより行われる。検出されたセグメントの座標は、座標メモリ20にセグメント座標テーブルとして格納される。

【0057】図4に座標メモリ20の記憶状態（内部状態）すなわちセグメント座標テーブルの一例を示す。図中左側に、文字ボタン「ム」すなわちセグメント $S_1$ を示し、図中右側に、セグメント $S_0$ 、 $S_1$ および $S_2$ のそれぞれの座標（図3の図中の $X_s$ 、 $X_e$ 、 $Y_s$ 、 $Y_e$ の各成分値の組で表される。）が記載されたセグメント座標テーブルを示す。例えば、セグメント $S_0$ の座標成分は、 $X_s$ が1、 $X_e$ が36、 $Y_s$ が1および $Y_e$ が84といった具合である。このように、セグメント抽出部38は、座標メモリ20のセグメント番号（例えば、セグメント $S_0$ を表す記号 $S_0$ の添字0のこと。）に対応した所定の格納場所に、そのセグメントの4頂点を座標成分値として読出し自在に記録する。

【0058】セグメントの抽出が終了すると、セグメント抽出部38は、セグメント統合部40に対して動作を開始するように、制御部24を介して指示を与える。

【0059】上述のセグメント統合部40は、セグメント抽出部38により抽出されたセグメント同士を、各々の位置座標に基づいて統合することにより新規セグメントを生成し、この新規セグメントの位置座標を上述の座標メモリ20に追加して記録する手段である。図5のブロック図に、セグメント統合部40の詳細な構成を示す。この実施の形態のセグメント統合部40は、第1読出部42、第2読出部44、距離検出部46、比較部48、セグメント生成部50および書込部52を具えている。

【0060】以下、図3に示す入力文字列ボタン34を処理対象とする場合のセグメント統合部40の動作につき説明する。図3の入力文字列ボタン34においては、セグメント $S_0$ とセグメント $S_1$ とが統合されて、新規セグメント $S_3$ が生成される。上述したように、この新規セグメントの生成は、セグメント抽出部38により抽出されたセグメント同士を各々の位置情報に基づいて統合することにより行われる。この統合処理を説明するに当たり、図6に示すフローチャートを参照する。

【0061】この例では、上述した制御部24が2つのカウンタを具えているとする。まず、制御部24の第1カウンタのカウント数 $n$ （ $n$ は整数）に初期値として0を代入し、また、制御部24の第2カウンタのカウント数 $k$ （ $k$ は整数）に初期値として1を代入する（図6のS9）。制御部24は、カウント数 $n$ に応じたセグメン

ト $S_n$ とカウント数 $k$ に応じたセグメント $S_k$ とを、座標メモリ20から読み出すように、第1および第2読出部42および44に対してそれぞれ制御信号を出力する。

【0062】そして、第1読出部42は、記録されたセグメント $S_n$ の位置情報を、座標メモリ20から読み出す（図6のS10）。この動作により、最初に、セグメント $S_0$ の位置情報が、座標メモリ20から読み出される。

【0063】続いて、第2読出部44は、入力文字列ボタン34中のセグメント $S_0$ に関して一定の側にあるセグメント $S_k$ の位置情報を、座標メモリ20から読み出す（図6のS11）。ここで、上述した「入力文字列ボタン34中のセグメント $S_0$ に関して一定の側」とは、入力文字列方向にセグメント番号が大きくなる向きを意味する。従って、セグメント $S_0$ に関しては、セグメント $S_1$ および $S_2$ が位置している入力文字列ボタン34の領域を指し、また、セグメント $S_1$ に関しては、セグメント $S_2$ が位置している入力文字列ボタン34の領域のことを指している。まず、この場合、セグメント $S_0$ に隣接するセグメント $S_1$ の位置情報が、第2読出部44により座標メモリ20から読み出される。

【0064】次に、距離検出部46は、セグメント $S_n$ とセグメント $S_k$ との間の距離 $D_{nk}$ を、読み出された各々の位置情報から求める（図6のS12）。すなわち、第1読出部42により読み出されたセグメント $S_0$ の位置情報と、第2読出部44により読み出されたセグメント $S_1$ の位置情報とが、距離検出部46に入力し、これらセグメント $S_0$ および $S_1$ 間の距離 $D_{01}$ が求められる。ここで、セグメント $S_0$ および $S_1$ 間の距離 $D_{01}$ は、図7に示すように、イメージ上において各セグメント $S_0$ および $S_1$ の始端位置（図4の図中左側に示す $X_s$ の位置に相当する。）同士を結ぶ入力文字列方向の直線の距離として定義される。例えば、図4に示すセグメント座標テーブルによれば、 $D_{01}=36$ となる。

【0065】このように、距離検出部46は、各セグメントの始端位置のX座標成分同士の差を検出することにより、この距離を検出する構成としてある。例えば、距離検出部46として、通常の差演算回路を用いることができる。

【0066】次に、距離検出部46により検出された距離 $D_{01}$ は、比較部48に伝送される。この比較部48は、入力文字列の行高さ $L$ の定数 $e$ 倍（ $e$ は正の実数）と、距離検出部46が求めた距離 $D_{nk}$ との比較を行う（図6のS13）。

【0067】ここで、行高さ $L$ は、入力文字列ボタン34を構成する各セグメントの入力文字列方向に垂直な方向（図3のY方向）の長さのことである。但し、この実施の形態では、入力文字列ボタン34を構成するセグメントのうち、上述の長さが最大のものを代表として選択

して、入力文字列パタンの行高さ $L$ と定義している。この入力文字列の行高さ $L$ には、座標メモリ20に格納された各セグメントの位置情報の中から、各 $Y$ 座標成分の差が最大となるものを、予め検出して記憶させておけばよい。例えば、図4に示すセグメント座標テーブルにおいては、 $L=87$ である。

【0068】また、この実施の形態では、定数 $e$ として、1.2を設定してある。この「1.2」という値は、繰返しテストを行い、経験的に定められた値である。この定数 $e$ の値は、比較部48が具える読出し自在のメモリ手段に、予めキーボード等の入力手段により設定しておく。そして、比較部48は、距離検出部46からの距離 $D_{nk}$ の入カタイミングで、上述のメモリ手段から設定した定数 $e$ の値が読み出されるように構成されている。

【0069】このように構成してあるので、比較部48は、入力される距離 $D_{nk}$ と、値1.2 $L$ との大小関係を求めることができる。そして、比較部48は、求めた大小関係に対応した信号を、セグメント生成部50に出力する。

【0070】次に、セグメント生成部50は、比較部48の出力信号が $D_{nk} \leq e \cdot L$ に応じた信号であるとき、セグメント $S_n$ とセグメント $S_k$ とを統合して新規セグメントを生成する(図6のS14)。上述の例では、 $D_{01}=36$ 、 $L=87$ である。従って、比較部48の比較結果が $D_{01} \leq e \cdot L$ であるから、セグメント生成部50が、第1および第2読出部42および44からセグメント $S_0$ およびセグメント $S_1$ の位置情報をそれぞれ入力し、これらセグメントの位置情報の統合を行う。図3に示すように、セグメント $S_0$ および $S_1$ が統合されることにより、新規セグメント $S_3$ が生成される(セグメント $S_3$ は、文字パタン「弘」に外接する矩形領域である。)

【0071】このようにして生成された新規セグメント $S_3$ は、セグメント $S_0$ とセグメント $S_1$ との両者を含む入力文字列パタン上の領域を、新たに設定することに等しい。図8に示すように、生成された新規セグメント $S_3$ の座標成分値には、セグメント $S_0$ とセグメント $S_1$ との対応する座標成分値のうちの、いずれか一方が選択されて設定される。そして、その選択は、生成される新規セグメントが、なるべく広い入力文字列パタン中の領域を占めるようになされる。例えば、 $X_3$ の座標成分については、セグメント $S_0$ が1、セグメント $S_1$ が37であり、新規セグメント $S_3$ の座標成分 $X_3$ として1が設定される。また、 $X_e$ の座標成分については、セグメント $S_0$ が36、セグメント $S_1$ が106であり、新規セグメント $S_3$ の座標成分 $X_e$ として106が設定される。

【0072】そして、書込部52は、セグメント生成部50により生成された新規セグメントの位置情報を座標

メモリ20に追加して記録する(図6のS15)。図8には、統合後のセグメント座標テーブルの様子を示す。このようにして、セグメント $S_0$ とセグメント $S_1$ との統合処理が完了する。

【0073】次に、制御部24は第2カウンタを参照して、セグメント $S_0$ に関して上述した側にあるセグメント( $S_1$ および $S_2$ )が、座標メモリ20に全て記録されたかどうかを調べる(図6のS16)。これは、抽出したセグメント番号の最大値を $p$ ( $p$ は整数。この例では、 $p=2$ 。)とすると、第2カウンタのカウント数 $k$ が $p$ になったかどうかを調べればよい。このときは、 $k=1$ であるから、 $k \neq p$ であり、制御部24は、次に、カウント数 $k$ に1を加えてカウントアップさせ(図6のS17)、第2読出部44にセグメント $S_2$ を読み出させる(図6のS11)。

【0074】そして、今度は、距離検出部46により、セグメント $S_0$ とセグメント $S_2$ との間の距離 $D_{02}$ を検出する(図6のS12)。この場合には、比較部48の比較結果は $D_{02} > e \cdot L$ である(図6のS13)。従って、これらセグメントは統合されない。

【0075】続いて、制御部24は、再び第2カウンタを調べるが(図6のS16)、このときカウント数 $k$ が2であるから $k=p$ である。これに回答して、制御部24は、第2カウンタのカウント数 $k$ を $(n+2)$ に設定する(図6のS18)。ここでは、カウント数 $n$ は0であるから、カウント数 $k$ は2に設定される。

【0076】次に、制御部24は、第1カウンタのカウント数 $n$ を調べる(図6のS19)。制御部24は、第1カウンタのカウント数 $n$ を参照することにより、 $n \neq p-1$ ということを確認する。そして、制御部24は、カウント数 $n$ に1を加えてカウントアップさせ(図6のS20)、次に、第1読出部42にセグメント $S_1$ の位置情報を読み出させる(図6のS10)。

【0077】そして、今度は、第2読出部44により、セグメント $S_1$ に関して上述した側にあるセグメント $S_2$ の位置情報を座標メモリ20から読み出す(図6のS11)。同様に、セグメント $S_1$ および $S_2$ 間の距離 $D_{12}$ を検出し(図6のS12)、距離 $D_{12}$ が $e \cdot L$ より大きいことを確認し(図6のS13)、第2カウントのカウント数 $k$ が $k=p$ であることを確認する(図6のS19)。このとき、 $n=1$ であるからカウント数 $k$ に形式的に3が設定されるが(図6のS18)、次に、第1カウンタのカウント数 $n$ が1であることが制御部24により確認されるので(図6のS19)、 $n=p-1$ となり、処理は終了する。

【0078】以上説明した通り、セグメント $S_0$ 、 $S_1$ および $S_2$ と新規セグメント $S_3$ とが取得できる。以下、これらセグメント $S_0$ 、 $S_1$ および $S_2$ と、新規セグメント $S_3$ とを併せて、各々を、候補文字パタンと称する。

【0079】＜文字認識＞次に、抽出された候補文字パタンの文字認識を行うステップにつき説明する（図2のS5）。このステップで行われる処理は、文字認識部16によりなされる。この実施の形態の文字認識部16は、候補文字ボタンに対応した文字コードであって、類似度の高い順に並べられた上位の一定個数の文字コードを、認識結果として、候補文字ボタンごとに文字コード格納部22に記録する手段である。この文字認識の結果、画像データである候補文字ボタンが文字コードに符号化される。

【0080】最初に制御部24が、セグメント統合部40からの動作終了の信号に応答して、座標メモリ20に格納されている候補文字の位置情報を読み出す。制御部24は、読み出した位置情報に基づいて、画像メモリ36に格納されている原画像（入力文字列）から、候補文字ボタンに対応した領域（文字画像と称する。）を切り出す。この文字画像の切出しは、上述した画像入力部10が具える切出し部で行ってもよいし、あるいは、制御部24がこのための切出し手段を具えていてもよい。切り出された文字画像は、評価系制御部30を介して、文字認識部16に伝送される。そして、文字認識部16は、取り込まれた文字画像に対して通常の文字認識処理を施す。従って、文字認識部16は、切り出された文字画像から特徴を抽出するための抽出手段と、標準文字の特徴を予め具えた辞書とを具えており、また、抽出手段で抽出された文字画像の特徴と標準文字の特徴とを比較するためのマッチング手段を具えている。このマッチング手段により、特徴間の類似度が算出される。そして、この実施の形態では、類似度の高い順に上位10個の文字コードが1つの候補文字ボタンに対して求められる。

【0081】ここで、類似度とは、認識対象の候補文字パタンの特徴と、認識によりこの候補文字ボタンから変換された文字コードの特徴との間の類似の度合いのことをいう。この実施の形態では、類似度として距離（相違度ともいう。上述したセグメント間の距離 $D_{nk}$ とは異なる。）を用いている。すなわち、いわゆる特徴ベクトル同士の近さのことである。類似度と距離とは逆比例関係にあるから、この実施の形態のマッチング手段は、距離の小さい順に上位10個の文字コードを取得するようになっている。尚、この実施の形態では、特徴間の距離を得る手法に対しては特に問わない。また、特徴を比較する際に用いる尺度についても任意に選択すればよい。

【0082】図9に、認識結果の一例を示す。図9の図中左側には、図3に示したセグメント $S_0$ を示し、図9の図中右側の表には、セグメント $S_0$ に対して得られた上位10個の文字コードを、その順位および距離値と対応付けて示してある。図9の対応関係に示すように、文字認識部16は、取得した文字コードと、その文字コードに対する距離値とを、距離の小さい順すなわち類似度の高い順に、セグメント番号（候補文字番号）に対応し

た格納場所（アドレス）の文字コード格納部22に記録する。

【0083】＜文字コードの評価＞次に、文字評価部32の動作につき説明する（図2のS6）。文字評価部32は、文字コード格納部22に記録されている文字コードにその文字種に応じた文字評価値を付与して、この文字評価値を文字コードに対応した文字コード格納部22の格納場所に格納する手段である。上述した通り、文字評価部32は、制約係数処理部26と、評価値計算部28と、評価系制御部30とをもって構成されており、これらが相俟って上述の動作を遂行する。この文字評価部32は、文字認識部16が文字認識動作を終了すると共に、動作を開始する。

【0084】そして、評価系制御部30は、文字コード格納部22に記録されている文字コードおよびその距離値を一定の規則で順次を読み出し、続いて、評価値計算部28に伝送する。例えば、候補文字番号順および距離値順に、各文字コードおよび距離値を読み出す。一方、評価値計算部28では、読み出した文字コードの文字種を判定し、その文字種に応じた制約係数が制約係数処理部26から入力されるように構成されている。上述したように、この実施の形態では、文字種として、記号・数字、カタカナ、平仮名および漢字の4種の文字種を想定しており、評価値計算部28は、読み出した文字コードを、これら4種の文字種に分類する。

【0085】この実施の形態の評価値計算部28は、文字評価値を、文字コードの類似度（距離）と予め設定した文字種に応じた制約係数との積として求めるための手段である。文字評価値は、候補文字ボタンから変換された文字コードが、入力文字列パタンを構成する文字として選択される確率を表したものとイえる。この実施の形態では、上述の文字種に応じた制約係数を、この文字種が入力文字列パタン中に出現する確率の値としてある。認識対象の入力文字列パタン34に対して設定された制約係数は、上述したように $\alpha=0$ 、 $\beta=1$ 、 $\gamma=1$ および $\eta=1$ であり、入力文字列パタン34には数字・記号の文字種が出現しないことを前提としている。

【0086】以上説明したように、評価値計算部28は、候補文字ボタンごとに、および距離値順に入力される文字コードおよび距離値から、先ず、その文字種を判別し、その文字種に応じた制約係数を制約係数処理部26から読み出す。そして、読み出した制約係数を距離値に積算する。この積算結果である文字評価値は、評価系制御部30を経て文字コード格納部22の対応した格納場所に格納される。

【0087】図10に、この文字評価処理を行った後の文字コード格納部22の内部状態を示す。図10において、文字コード（図中のコードの欄）と、距離と、制約係数（図中の係数の欄）と、文字評価値（図中の評価の欄）との関係を、各候補文字番号ごとに4つの表にて示



してある。図10(A)に候補文字ボタン $S_0$ の評価結果を示し、図10(B)に候補文字ボタン $S_1$ の評価結果を示し、図10(C)に候補文字ボタン $S_2$ の評価結果を示し、および図10(D)に候補文字ボタン $S_3$ の評価結果を示す。この実施の形態では、類似度として距離を用いているので、文字評価値が小さい程、その文字コードが入力文字列中に出現する文字である確率が大きい。

【0088】ところで、制約係数が0の場合には文字評価値も0となるが、制約係数が0の文字コードは入力文字列中に登場しない文字であるから、この表に示されるように、これをNULL(空白文字)として扱う。このように、以降の処理においては制約係数が0(この実施の形態では、文字評価値が0としても同様)の文字コードをNULLとして扱う。従って、入力文字列を構成する文字コードの候補数が削減される。例えば、図10において、候補文字ボタン $S_0$ に対して求められた文字コードの中には、記号・数字の文字種である文字コードが4つあり、これらの文字コードの文字評価値にはNULLが設定されている。

【0089】以上説明したように、文字評価部32は、文字評価値として文字種に応じた制約係数を用いて処理を行い、この制約係数を、この文字種が入力文字列に出現するか否かを「1」または「0」のそれぞれ2値で表した値として処理を行う手段である。

【0090】＜候補文字コード列の作成＞次に、候補文字コード列作成部14の動作につき説明する(図2のS7)。候補文字コード列作成部14は、座標メモリ20に記録された位置情報を利用して複数の候補文字コード列を作成する手段であり、座標メモリ20に記録されている位置情報と、文字コード格納部22に記録されている文字評価値とを参照して、候補文字コード列を作成する。図11のブロック図に候補文字コード列作成部14の詳細な構成を示す。図11に示すように、候補文字コード列作成部14は、テーブル作成部54と、テーブルメモリ部56と、処理回路58と、候補文字記録部60と、文字番号格納部62と、文字コード列格納部64とを具えている。また、図示していないが、処理回路58は、指定した始点位置を記録するためのメモリ手段を具えている。

【0091】まず、評価系制御部30は、文字評価の処理が終了したという信号を制御部24に出力する。制御部24は、評価系制御部30からの信号に応答して、候補文字コード列の作成処理を開始する。最初に制御部24は、テーブル作成部54の動作を開始させる。

【0092】テーブル作成部54は、記録された候補文字ボタンの位置情報を座標メモリ20から読み出して、この位置情報に基づき各候補文字ボタンの始点位置および終点位置を切出し候補位置 $C_i$ ( $i$ は整数)として求め、これら候補文字ボタンと切出し候補位置との対応関

係をテーブルメモリ部56に記録する手段である。

【0093】入力文字列ボタン34からの文字の切出しは、X軸上の2点をそれぞれ始点位置および終点位置として指定することにより行う。テーブル作成部54は、上述したように、入力文字列ボタンから文字を切り出すために指定が可能な全ての位置を切出し候補位置として指定する。図3に、入力文字列ボタン34の切出し候補位置を示す。入力文字列ボタン34においては、4つの切出し候補位置 $C_0$ 、 $C_1$ 、 $C_2$ および $C_3$ が設定できる。この実施の形態では、切出し候補位置は、主走査方向に順序付けられて番号付けされている。これら切出し候補位置の中から2つを選択することにより、任意の候補文字番号を指定することができる。例えば、切出し候補位置 $C_0$ と切出し候補位置 $C_1$ とを選択することにより、 $C_0$ を始点位置とし、 $C_1$ を終点位置とする候補文字番号 $S_0$ が指定できる。また、切出し候補位置 $C_0$ と切出し候補位置 $C_2$ とを選択することにより、 $C_0$ を始点位置とし、 $C_2$ を終点位置とする候補文字番号 $S_3$ が指定できる。

【0094】上述したように、切出し候補位置は、入力文字列ボタンの両端位置と、隣接する候補文字ボタン間の境界位置として抽出される。この実施の形態のテーブル作成部54は、切出し候補位置 $C_0$ 、 $C_1$ および $C_2$ を候補文字ボタン(セグメント) $S_0$ 、 $S_1$ および $S_2$ の各始端位置(すなわち図4に示す座標成分 $X_s$ )の読出しにより抽出し、また、切出し候補位置 $C_3$ を入力文字列ボタンの最後尾のセグメント $S_2$ の終端位置(すなわち図4に示す座標成分 $X_e$ )の読出しにより抽出する。

【0095】次に、テーブル作成部54は、設定した切出し候補位置と候補文字との対応関係を求めて、この対応関係をテーブルメモリ部56に記録する。図12に、この対応関係が記録されたテーブルメモリ部56の内部状態を、セグメントテーブルとして示す。このセグメントテーブルは、グラフ理論の分野で通常に用いられる隣接行列として表されている。図中において、行欄の項目枠には始点位置としての切出し候補位置番号 $C_i$ が記載されており、列欄の項目枠には終点位置としての切出し候補位置番号 $C_j$ が記載されている。そして、行と列の交差部分のデータ枠には、始点位置と終点位置とに対応した候補文字番号 $S_n$ が記載されている(尚、空白のデータ枠にはNULLが設定されている)。従って、始点位置としての切出し候補位置番号と、終点位置としての切出し候補位置番号とを指定することにより、この表から、切出し対象の候補文字番号が指定できるようになっている。入力文字列ボタンの適当な切出し位置を決定するためには、次に説明する処理回路58において、この表を活用した処理を行う。

【0096】処理回路58は、テーブルメモリ部56に記録された対応関係(セグメントテーブル)を参照して



作成関数  $F(C_i, P)$  を用いた処理を行うことにより候補文字番号列を作成し、この候補文字番号列と文字コード格納部22に記録された文字評価値とを参照して候補文字コード列を作成する手段である。

【0097】ここで、上述の候補文字番号列は、候補文字パタンの配列情報であり、候補文字番号の配列として表される。また、上述の候補文字コード列は、文字コードの配列情報である。この処理回路58では、最初に、候補文字番号列の作成を行い、この候補文字番号列に従い最適な文字コードの配列を候補文字コード列として作成する。また、作成関数  $F(C_i, P)$  の引き数  $P$  は、読み出した候補文字番号を、読出し順に対応させて候補文字記録部60に記録しておくためのデータファイルであり、以下、このファイルをパスと称している。

【0098】この実施の形態の処理回路58は、作成関数  $F(C_i, P)$  を用いた処理は、グラフ探索のアルゴリズムに基づいたものである。処理回路58は、上述したテーブルメモリ部56、候補文字記録部60および文字番号格納部62を記憶手段として用いて処理を行う。以下、この作成関数  $F(C_i, P)$  による処理手順につき、図13のフローチャートを参照して説明する。

【0099】処理回路58は、2つのカウンタを具えており、始めに、始点位置を指定するための第1カウンタのカウント数  $i$  に0を設定しておく（図13のS21）。また、終点位置を指定するための第2カウンタのカウント数  $j$  には、1を設定しておく（図13のS21）。さらに、パス  $P$  は初期化して  $P = \Phi$ （ $\Phi$  はデータが無いということを表す記号）としておく（図13のS21）。以下に説明する（e1）、（e2）、（e3）および（e4）のステップ（図13のS22～S33）が、作成関数  $F(C_i, P)$  により実行される。

【0100】ステップ（e1）：処理回路58は、テーブルメモリ部56に記録した切出し候補位置を始点位置  $C_i$  として指定する（図13のS22）。このとき、カウント数  $i$  には0が設定されているから、切出し候補位置  $C_0$  が始点位置として指定される。また、この始点位置  $C_0$  を指定順に上述のメモリ手段に記録しておく（図13のS23）。以下、このメモリ手段に始点位置  $C_0$  が記録されていることを、 $R = C_0$  で表す。

【0101】ステップ（e2）：指定した始点位置  $C_0$  が入力文字列パタンの最終端位置か否かを判別する（図13のS24）。このステップ（e2）の判別結果に応じて、以下のステップ（e3）またはステップ（e4）のいずれか一方のステップに進む。入力文字列パタン34の最終端位置すなわち最右端位置は、切出し候補位置  $C_3$  である。従って、位置  $C_0$  は最終端位置ではないので、次にステップ（e3）に進む。

【0102】ステップ（e3）：始点位置  $C_i$  が最終端位置ではない場合には、始点位置  $C_i$  に対して指定が可

能なテーブルメモリ部56に記録されている切出し候補位置を終点位置  $C_j$  として指定する（図13のS25）。ここでは、カウント数  $j$  に1が設定されているから、終点位置として切出し候補位置  $C_1$  が指定される。この実施の形態では、通常は、指定位置に隣接する切出し候補位置が終点位置として指定される。

【0103】次に、テーブルメモリ部56に記録されている対応関係から、指定した始点位置  $C_i$  および終点位置  $C_j$  により候補文字番号を指定して、この候補文字番号（図13には記号  $S_k$  で表す。）をパス  $P$  に格納する（図13のS26）。つまり、 $C_i$  および  $C_j$  間の候補文字番号をパス  $P$  に加える。ここでは、 $C_0$  および  $C_1$  間の候補文字番号  $S_0$  をパス  $P$  に記録する（図12）。パス  $P$  に候補文字番号  $S_0$  が記録されていることを  $P = S_0$  で表す。

【0104】上述した通り、処理回路58は、読み出した切出し候補位置  $C_0$  および  $C_1$  間の候補文字番号  $S_0$  を、候補文字記録部60に順次に配列情報として格納する。候補文字記録部60は、候補文字番号を、入力された順に配列させて記憶する。例えば、次のステップにおいて候補文字番号  $S_1$  が指定された場合には、候補文字記録部60に候補文字番号  $S_0$  および  $S_1$  が番号順に順序付けられて記憶されることになる。尚、例えば、候補文字記録部60には、候補文字番号  $S_k$  に対応したアドレス（格納場所を指定するための情報）が、配列順に記録されるとしてもよい。

【0105】次に、作成関数  $F(C_j, P)$  の処理を開始する（図13のS27）。ここでは、作成関数  $F(C_i, P)$  が呼び出される。このように、作成関数  $F(C_0, P)$  は再帰的に作成関数  $F(C_1, P)$  の呼び出しを行う。このように、再帰的に作成関数を呼び出して、配列情報  $P$  を完成させる。ここで、配列情報  $P$  の完成とは、選択されて格納された候補文字番号をもって、処理対象としている入力文字列パタン34が再現されるようになることを意味する。以下、作成関数  $F(C_i, P)$  の処理を実行する。

【0106】作成関数  $F(C_i, P)$  は、前述したステップ（e1）から処理を開始し、このとき、始点位置として切出し候補位置  $C_i$  が指定される（図13のS22）。また、この始点位置  $C_i$  を指定順に処理回路58が具えるメモリ手段に記録する（図13のS23）。この時点で、メモリ手段には  $C_0$  および  $C_1$  が、この順に記録されている。従って、 $R = C_0, C_1$  である。

【0107】次に、切出し候補位置  $C_i$  が最終端位置か否かを調べる（図13のS24）。切出し候補位置  $C_i$  は、最終端位置ではないので、次に、終点位置の指定を行う。終点位置としては、切出し候補位置  $C_j$  が指定される（図13のS25）。そして、始点位置  $C_i$  および終点位置  $C_j$  間の候補文字番号  $S_1$  をパス  $P$  に加え（図13のS26）、次に、作成関数  $F(C_j, P)$  の呼び

出しを行う(図13のS27)。バスPには、候補文字番号 $S_0$ および $S_1$ が順序付けられて格納されており、このことを $P=S_0 S_1$ と表す。

【0108】次に、作成関数 $F(C_2, P)$ の処理が開始され、始点位置として切出し候補位置 $C_2$ が指定される(図13のS22)。また、この始点位置 $C_2$ をメモリ手段に記録し、 $R=C_0 C_1 C_2$ とする(図13のS3)。

【0109】そして、切出し候補位置 $C_2$ が最終端位置ではないことを判別し(図13のS24)、終点位置として切出し候補位置 $C_3$ を指定する(図13のS25)。よって、位置 $C_2$ および $C_3$ 間の候補文字番号 $S_2$ がバスPに加えられる(図13のS26)。次に、作成関数 $F(C_3, P)$ の呼び出しを行う(図13のS27)。このとき、 $P=S_0 S_1 S_2$ である。

【0110】次に、作成関数 $F(C_3, P)$ の処理が開始され、始点位置として切出し候補位置 $C_3$ が指定される(図13のS22)。この始点位置 $C_3$ は、メモリ手段に記録され、 $R=C_0 C_1 C_2 C_3$ となる(図13のS23)。次に、この位置 $C_3$ は最終端位置であるから、ステップ(e4)に進む(図13のS24)。

【0111】ステップ(e4)：始点位置 $C_1$ が最終端位置である場合には、バスPを文字列格納部62に保存する(図13のS29)。この実施の形態では、バスPの保存を行う前に、 $C_0$ および $C_3$ 間の候補文字番号が入力文字列を再現せしめるようにバスPに格納されている、バスPが空でないことを確認する(図13のS28)。この場合のバスPには空でないので、「 $S_0 S_1 S_2$ 」の配列情報が第1の候補文字番号列として、文字番号格納部62に保存される。

【0112】次に、全候補文字番号が文字番号格納部62に記録されたか否かを判別する(図13のS30)。この判別結果により、全候補文字番号が文字番号格納部62に記録された場合には処理が終了する。この例では、未だ、文字番号格納部62に候補文字番号 $S_3$ が記録されていないので、メモリ手段を参照して、始点位置 $C_1$ より2つ前に記録されている切出し候補位置 $C_1$ と最終端位置との間の候補文字番号をバスPから消去する(図13のS31)。この時点では、メモリ手段に $R=C_0 C_1 C_2 C_3$ が格納されている。従って、始点位置 $C_3$ より2つ前に記録されている位置 $C_1$ が位置 $C_1$ となり、よって、位置 $C_1$ および $C_3$ 間の候補文字番号 $S_1$ および $S_2$ を、候補文字記録部60のバスPから消去する。この結果、 $P=S_0$ となる。また、メモリ手段から、切出し候補位置 $C_1$ ( $=C_1$ )と、最終端位置 $C_3$ と、位置 $C_1$ および $C_3$ 間の切出し候補位置 $C_2$ とを、消去する(図13のS32)。よって、 $R=C_0$ である。そして、カウント数 $j$ を $f$ として(図13のS33)、次に、関数 $F(C_1, P)$ すなわち $F(C_1, P)$ の呼び出しを行う(図13のS27)。

【0113】次に、作成関数 $F(C_1, P)$ の実行により、まず、始点位置として $C_1$ を指定し(図13のS22)、この始点位置 $C_1$ をメモリ手段に記録して、 $R=C_0 C_1$ とする(図13のS23)。続いて、この位置 $C_1$ が最終端位置ではないことが判別される(図13のS24)。次に、カウント数 $j$ には2が設定されているから、通常であれば終点位置として切出し候補位置 $C_2$ が指定されるが、先程指定した切出し候補位置 $C_1$ (このとき $C_1=C_1$ )と最終端位置 $C_3$ との間の位置は指定しないようにしてあるため、ここでは、 $C_2$ に隣接する $C_3$ が指定される(図13のS25)。そして、位置 $C_1$ および $C_3$ で指定される候補文字番号は存在しないので、次の処理により、バスPには候補文字番号の代わりに $\Phi$ が設定される(図13のS26)。このとき $P=S_0 \Phi$ と表せる。次に、作成関数 $F(C_3, P)$ を呼び出す(図13のS27)。

【0114】そして、次に指定する始点位置 $C_3$ は、最終端位置であるから、次に、バス中に $\Phi$ が存在するか否かを判別する(図13のS28)。この場合には、 $P=S_0 \Phi$ であるから、バスP中に $\Phi$ が存在し、従って、バスPは保存されない。次に、全候補文字番号が文字番号格納部62に格納されたか否かを確認する(図13のS30)。この時点で、文字番号格納部62には、全部の候補文字番号が格納されていないため、上述したのと同様に、 $C_1$ (このとき $R=C_0 C_1 C_3$ であるから $C_1=C_0$ である。)以降の候補文字番号をバスPから消去して $P=\Phi$ とし(図13のS31)、 $C_1$ および $C_3$ 間の切出し候補位置をメモリ手段から消去して $R=\Phi$ とする(図13のS32)。そして、カウント数 $j$ を $f$ ( $=0$ )にして(図13のS33)、次に、作成関数 $F(C_0, P)$ の呼び出しを行う(図13のS27)。

【0115】以下同様にして、始点位置 $C_0$ の指定および記録( $R=C_0$ )、終点位置 $C_2$ の指定、 $C_0$ および $C_2$ 間の候補文字番号 $S_3$ のバスPへの記録( $P=S_3$ )、作成関数 $F(C_2, P)$ の呼び出し、始点位置 $C_2$ の指定および記録( $R=C_0 C_2$ )、終点位置 $C_3$ の指定、 $C_2$ および $C_3$ 間の候補文字番号 $S_2$ のバスPへの記録( $P=S_3 S_2$ )、作成関数 $F(C_3, P)$ の呼び出し、バスPの保存(配列情報「 $S_3 S_2$ 」が第2の候補文字番号列として保存)を行い、全候補文字番号が文字番号格納部62に記録されたことの確認を行って処理を終了する。以上説明した処理により、文字番号格納部62には、2つの候補文字番号列が格納される。

【0116】＜作成関数の別の説明＞以上説明した作成関数の処理は、図26に示すフローチャートによって簡潔に表現することができる。図26は、作成関数 $F(C_i, P)$ のグラフ探索フローを示す図である。また、図27の表は、切出し候補位置 $C_1$ 、 $C_2$ および $C_3$ とセグメント $S_1$ 、 $S_2$ および $S_3$ との対応関係を示したセグメントテーブルである。以下、作成関数 $F(C_i,$

P)を、図27に示すセグメントテーブルを処理対象とする場合につきステップごとに説明する。尚、以下のインデントは、再帰呼出しのレベルを示している。

【0117】[ステップ0] 先ず、候補文字記録部60はクリアしておき、 $P=\Phi$ としておく。そして、作成関数 $F(C_1, \Phi)$ を呼び出す(Start of  $F(C_1, \Phi)$ )。

【0118】[ステップ1] 次に、切出し候補位置 $C_1$ が最右端か否かを判別する(図26のS40)。切出し候補位置 $C_1$ は、最右端ではないので、ステップ2へ進む。

【0119】[ステップ2] 切出し候補位置 $C_1$ から辿れる全ての切出し候補位置 $C_j$ につき以下の処理を行う(図26のS41)。尚、図26に示すループ1の処理(図27のS40およびS46間の処理)は、切出し候補位置 $C_i$  ( $i$ は整数)の右側に位置する切出し候補位置 $C_j$  ( $j$ は $j > i$ を満たす整数)のすべてが読み出されるまで続行される(For each  $C_j \in (C_i, C_j) \neq \text{NULL}$ )。

【0120】[ステップ3]  $C_1$ と $C_2$ 間にあるセグメント $S_1$ を検出する(図26のS42:  $S_{k+1} \leftarrow \text{Adj}(C_i, C_j)$ )。

【0121】[ステップ4]  $S_1$ を候補パスPへ追加し、 $P=S_1$ とする(図26のS43)。

【0122】[ステップ5]  $C_j$ およびPを引き数とする作成関数すなわち作成関数 $F(C_2, S_1)$ を呼び出す(図26のS44)。

【0123】[ステップ5-1]  $C_2$ は最右端ではないので、次に、ステップ5-2へ進む(図26のS40)。

【0124】[ステップ5-2] 切出し候補位置 $C_2$ から辿れる全ての切出し候補位置 $C_j$ につき以下の処理を行う(図26のS41)。

【0125】[ステップ5-3]  $C_2$ と $C_3$ 間にあるセグメント $S_2$ を検出する(図26のS42)。

【0126】[ステップ5-4]  $S_2$ を候補パスPへ追加し、 $P=S_1 S_2$ とする(図26のS43)。

【0127】[ステップ5-5] 次に、作成関数 $F(C_3, S_1 S_2)$ を呼び出す(図26のS44)。

【0128】[ステップ5-5-1]  $C_3$ は最右端であるので、候補文字記録部60中に記録されている候補パス $P=S_1 S_2$ を文字番号格納部62へ記録する(図26のS47)。

[ステップ5-5-2] 呼出し元に戻る。

【0129】[ステップ5-6]  $S_2$ を候補パスPから消去し、 $P=S_1$ とする(図26のS45)。

【0130】[ステップ6]  $S_1$ を候補パスPから消去する(図26のS45)。

【0131】[ステップ7]  $C_1$ と $C_3$ 間にあるセグメント $S_3$ を検出する(図26のS42)。

【0132】[ステップ8]  $S_3$ を候補パスPへ追加し、 $P=S_3$ とする(図26のS43)。

【0133】[ステップ9] 次に、作成関数 $F(C_3, S_3)$ を呼び出す(図26のS44)。

【0134】[ステップ9-1]  $C_3$ は最右端であるので、候補文字記録部60中に記録されている候補パス $P=S_3$ を文字番号格納部62へ記録する(図26のS47)。

[ステップ9-2] 呼出し元に戻る。

【0135】[ステップ10]  $S_3$ を候補パスPから消去し、 $P=\Phi$ とする(図26のS45)。

【0136】[ステップ11] すべての切出し候補位置 $C_j$ について処理が終了したので、グラフ探索フローを終了する。

【0137】以上説明した処理により、文字番号格納部62には、2つの候補文字番号列( $S_1 S_2$  および  $S_3$ )が記録される。

【0138】図14に、入力文字列ボタン34に対して作成関数の処理を行った後の文字番号格納部62の内部状態を候補文字番号列テーブルとして示す。図中の表には、第1および第2の候補文字番号列 $P_0$  および  $P_1$ が候補文字番号 $S_0$ 、 $S_1$ 、 $S_2$  および  $S_3$ の配列で示してある。第1の候補文字番号列 $P_0$ は、候補文字番号 $S_0$ 、 $S_1$  および  $S_2$ の配列を表す情報として、文字番号格納部62に記録されている。また、第2の候補文字番号列 $P_1$ は、候補文字番号 $S_3$  および  $S_2$ の配列を表す情報として、文字番号格納部62に記録されている。尚、図中において、空白欄には、NULLが設定されている。

【0139】処理回路58は、作成関数による処理を終了すると、次に、候補文字コード列を作成する処理に移る。すなわち、作成された候補文字番号列と文字コード格納部22に記録された文字評価値とを参照して候補文字コード列を作成する。

【0140】先ず、処理回路58は、文字番号格納部62に格納されている候補文字番号列を順次に1つずつ読み出す。そして、候補文字番号列に従った候補文字番号順に、候補文字番号で指定されるアドレスに格納された文字コードを、文字コード格納部22から読み出す。読み出された文字コードは、候補文字番号ごとに文字コード列格納部64に記録される。従って、文字コード列格納部64には、候補文字番号列ごとに、複数通りの文字コードの組合せが候補文字コード列として記録される。

【0141】例えば、候補文字番号列 $P_0$ について見れば、候補文字番号列 $P_0$ を構成する最初の候補文字番号は $S_0$ である。この候補文字番号 $S_0$ に対応した文字コード格納部22の格納場所に格納されている文字コードおよび文字評価値の組が順次に読み出されて、文字コード列格納部64に伝送される。これと同様にして、候補文字番号 $S_1$  および  $S_2$ に対応した文字コードが順次に

文字コード列格納部64に伝送されて記録される。候補文字番号列P<sub>0</sub>に属する候補文字コード列の一例として、「多、そ、三」等が挙げられる。

【0142】ところで、この実施の形態では、処理回路58は、文字評価値としてNULLが設定されている文字コードは読み出さないように構成されている。上述したように記号・数字の文字種である文字コードに対しては、NULLが設定されているので、例えば、候補文字番号S<sub>0</sub>として文字コード「3」は、処理回路58によって読み出されない。図10に示す文字コード格納部の内部状態を示す各表によれば、候補文字番号S<sub>0</sub>、S<sub>1</sub>およびS<sub>3</sub>としてそれぞれ10個の文字コードが記録されており、候補文字番号S<sub>2</sub>としては9個の文字コードが記録されている。しかしながら、図10によれば、候補文字番号S<sub>0</sub>に属する4つの文字コードに対してNULLが設定されており、候補文字番号S<sub>1</sub>に属する1つの文字コードに対してNULLが設定されている。従って、通常であれば、候補文字番号列P<sub>0</sub>に属する候補文字コード列として、900個の候補文字コード列が作成されるが、この実施の形態の場合には、486個の候補文字コード列だけが作成される。このように、この実施の形態の文字切出し装置によれば、候補文字コード列を通常よりも少なくすることができる。従って、以下に説明する候補文字コード列の選択の段階において、処理量を格段に減少させることが可能になる。

【0143】図15は、作成される候補文字コード列の一例を、入力文字列ボタン34と共に示したものである。図中上側部分に入力文字列ボタン34を示し、図中下側部分に候補文字コード列の一部を2端子有向グラフとして示してある。図15に示すように、候補文字コード列は、切出し候補位置間を結ぶ矢印により表されたパス（経路）である。図中の各矢印の下側に、矢印で結ばれる切出し候補位置により指定される候補文字に対して求められた文字コードとその文字評価値とを示してある。最良文字コード列選択部18では、切出し候補位置C<sub>0</sub>から切出し候補位置C<sub>3</sub>に亘って文字評価値を加算したときに、加算結果が最も小さくなるパスすなわち類似度が最も大きくなるパスが最良文字コード列として選択される。

【0144】次に、処理回路58は、候補文字コード列の作成を終えたことを制御部24に伝達し、これを受けて制御部24は、評価系制御部30を介して最良文字コード列選択部18に動作の開始を指示する。

【0145】＜最良文字コード列の選択＞このステップの処理は、最良文字コード列選択部18により行われる（図2のS8）。最良文字コード列選択部18は、文字コード格納部22から読み出された文字評価値を、作成した候補文字コード列に従い加算し、この加算の結果により類似度が最大となる候補文字コード列を最良文字コード列として選択する手段である。従って、最良文字コ

ード列選択部18は、文字コード列格納部64に記録されている候補文字コード列を参照して、その候補文字コード列に従い文字コード格納部22に格納されている文字評価値を読み出してゆき、1つの候補文字コード列に亘って加算を行う。そして、最良文字コード列選択部18は、この加算結果により類似度が最大となる候補文字コード列を最良文字コード列として選択して外部に出力する。

【0146】この実施の形態では、距離に基づいた文字評価値を用いているから、加算結果が最小の候補文字コード列が、類似度が最大の候補文字コード列である。図15に示した例によれば、入力文字列ボタン34に対しては、「弘、三」という候補文字コード列が、類似度が最も大きくなる候補文字コード列である。従って、この候補文字コード列が最良文字コード列として選択され、外部のコンピュータ装置等に出力される。このように最良文字コード列が決定されたため、入力文字列ボタン34の切出し位置は上述の切出し候補位置C<sub>0</sub>、C<sub>2</sub>およびC<sub>3</sub>として決定される。

【0147】以上説明した通り、この実施の形態の文字切出し装置は、文字評価部32を構成する制約係数処理部26、評価値計算部28および評価系制御部30が相俟って動作することにより、文字コードの評価を行って制限を課し、この制限を候補文字コード列を作成する段階で活用することにより、装置の処理量（演算量）を通常よりも少なくすることができる。

【0148】〔第2の実施の形態〕以下、第2の実施の形態につき説明する。尚、第1の実施の形態と重複する構成については同様の番号を付して示し、また、重複する事項については説明を省略する場合がある。

【0149】図16は、第2の実施の形態の文字切出し装置の構成を示すブロック図である。この実施の形態の文字切出し装置は、第1の実施の形態と同様に、画像入力部10、候補文字ボタン抽出部12、候補文字コード列作成部14、文字認識部16、最良文字コード列選択部18、座標メモリ20、文字コード格納部22、制御部24および文字評価部32を具えている。しかしながら、この実施の形態の文字評価部32は、第1の実施の形態の構成と異なり、制約係数処理部26および評価系制御部30は具えているが、評価値計算部28を具えていない。また、第2の実施の形態の文字切出し装置は、単語照合部66を具えている。この単語照合部66は、評価系制御部30により制御されており、最良文字コード列の選択時に用いられる。

【0150】以下、第2の実施の形態の動作につき説明してゆくが、理解を容易にするために、図17に示す入力文字列ボタンを認識対象とする場合につき説明する。図17には、「小川」という姓名を表す入力文字列ボタンが示されている。

【0151】先ず、第1の実施の形態と同様に、制約係

数の入力を行う(図2のS1)。この実施の形態では、制約係数を、その文字種が入力文字列に出現するか否かを「1」または「0」の2値で表した値として設定する。第1の実施の形態と同様に、文字種として、記号・数字、カタカナ、平仮名および漢字の4種を想定している。そして、図17に示すように、この実施の形態では、姓名を表す文書領域を認識対象とするため、制約係数として、記号・数字、カタカナおよび平仮名に対しては0を設定し、漢字に対してだけ1を設定する。これは、一般に、姓名としては、漢字が用いられることに基づいている。

【0152】次に、第1の実施の形態と同様にして、画像入力部10は、入力文字列ボタンを含む原画像を読み取り、画像メモリ36に記録する(図2のS2)。

【0153】次に、第1の実施の形態と同様にして、候補文字ボタン抽出部12は、入力文字列ボタンから候補文字ボタンの抽出を行う。候補文字ボタン抽出部12は、セグメント抽出部38と、セグメント統合部40とを具えている。まず、セグメント抽出部38により入力文字列ボタンからセグメントを抽出する(図2のS3)。この結果、セグメントS0、S1、S3、S4およびS5が抽出される。この抽出結果を、図18のセグメント座標テーブル(座標メモリ20の内部状態)に示す。次に、セグメント統合部40により、抽出されたセグメント同士の統合を行い、新規セグメントS6、S7、S8、S9、S10、S11、S12、S13およびS14が生成される(図2のS4)。この統合結果を、図19のセグメント座標テーブル(座標メモリ20の内部状態)に示す。

【0154】次に、第1の実施の形態と同様にして、文字認識部16により、候補文字ボタン抽出部12により抽出された候補文字ボタンS0～S14の文字認識を行う。図20に、候補文字ボタンS0の認識結果の一例を示す。この実施の形態では、類似度の高い順に上位10個の文字コードを順序付けて文字コード格納部22に記録するが、各文字コードの距離値は記録していない。

【0155】次に、文字認識部16により求めた文字コードの評価を行う(図2のS6)。このステップは、文字評価部32により行われる。まず、評価系制御部30は、文字コード格納部22に記録されている文字コードを順次に読み出し、その文字種を判別する。そして、評価系制御部30は、文字種に応じた制約係数を、制約係数処理部2.6から読み出し、文字コード格納部22の対応した格納場所に記録する。この実施の形態では、制約係数そのものを文字評価値として後の処理において用いる。

【0156】図21から図23に、各候補文字ボタンの評価結果を示す。尚、一部の候補文字ボタンについては省略してある。図21(A)に候補文字ボタンS0の評価結果を示し、図21(B)に候補文字S1の評価結果

を示し、図21(C)に候補文字ボタンS2の評価結果を示し、図21(D)に候補文字ボタンS3の評価結果を示しており、図22(E)に候補文字ボタンS4の評価結果を示し、図22(F)に候補文字ボタンS5の評価結果を示し、図22(G)に候補文字ボタンS6の評価結果を示し、図22(H)に候補文字ボタンS7の評価結果を示しており、図23(I)に候補文字ボタンS11の評価結果を示し、図23(J)に候補文字ボタンS12の評価結果を示し、図23(K)に候補文字ボタンS13の評価結果を示し、および図23(L)に候補文字ボタンS14の評価結果を示す。

【0157】次に、第1の実施の形態と同様にして、候補文字コード列の作成を、候補文字コード列作成部14により行う(図2のS7)。この実施の形態の候補文字コード列作成部14の構成は、図11のブロック図に示した構成と同様である。最初に、候補文字コード列作成部14によりセグメントテーブルが作成される。図24に、処理回路58およびテーブル作成部54により、テーブルメモリ部56に記録されたセグメントテーブルの様子を示す。このセグメントテーブルを参照して、処理回路58は、候補文字番号列を作成して文字番号格納部62に格納する。そして、処理回路58は、文字番号格納部62に格納された候補文字番号列と、文字コード格納部22に格納された文字評価値(制約係数)とを参照して、候補文字コード列を作成する。この実施の形態の処理回路58も、第1の実施の形態と同様に、文字評価値として「0」が付与されている文字コードは読みださない。よって、この実施の形態の文字切出し装置では、通常の文字切出し装置に比べて、少ない候補文字コード列が作成される。従って、処理量が少なくて済む。図25に、この実施の形態で作成される候補文字コード列の一部を2端子有向グラフとして示す。図25には、各矢印の下側に、その矢印が結合する切出し候補位置で指定される候補文字に対応した文字コードと、その文字コードの類似度順位とを、組にして示してある。

【0158】次に、最良文字コード列の選択を行う(図2のS8)。この実施の形態では、最良文字コード列の選択を、最良文字コード列選択部18と、単語照合部66とにより行う。この実施の形態の最良文字コード列選択部18は、最初に、候補文字コード列作成部14で作成された候補文字コード列と単語辞書との照合を行い該当する単語の有無を調べる。上述の単語辞書は、単語照合部66に具えられている。まず、最良文字コード列作成部18は、文字コード格納部64を参照して、文字コード格納部22から文字コードを読み出して入力文字列を構成し、単語照合部66において、単語辞書との照合を行う。この照合により、該当する単語が1つだけあった場合には、最良文字コード列としてその単語を選択すればよい。それ以外の場合には、最良文字コード列作成部18は、この照合結果に応じて、以下の処理①およ

び②のいずれか一方を行う。

【0159】① 該当した単語が複数あった場合には、それら該当単語を構成する文字コードの類似度の順位の合計が最も小さくなる候補文字コード列を最良文字コード列として選択する。

【0160】例えば、図25に示す候補文字コード列を、該当する単語があったものであるとすると、この中で順位の合計が最小となるもの、すなわち、類似度の合計が最も大きくなるものは、「小、川」の候補文字コード列である。従って、この候補文字コード列が最良文字コード列として選択される。

【0161】② 該当した単語が無かった場合には、候補文字コード列を構成する文字コードの類似度の順位の合計が最も小さくなる候補文字コード列を最良文字コード列として選択する。

【0162】このように、該当した単語が無かった場合には、最良文字コード列作成部18は、文字コード列格納部64に格納されている全ての候補文字コード列について、上述と同様に、各々の類似度順位の合計を算出する。

【0163】このようにすると、最良文字コード列選択部18は、最良文字コード列を選択することができる。

【0164】

【発明の効果】以上説明した通り、この発明の文字切出し方法によれば、各候補文字パタンの文字コードを類似度に基づいて取得し、取得した文字コードにその文字種に応じた文字評価値を付与し、その文字評価値を参照して候補文字コード列を作成するといった各ステップを含んでいる。そして、上述の文字評価値を、例えば、取得した文字コードの文字種が、1) 入力文字列に出現する文字種であるか、2) 入力文字列に出現しない文字種であるかに基づく値とすることにより、この2)に相当する文字評価値が付与された文字コードを、候補文字コード列を作成する段階で除外することができる。よって、探索空間の削減が図れる。

【0165】また、この発明の文字切出し方法の好適な実施例によれば、セグメントを抽出し、抽出した各セグメントの統合を行うことにより新規セグメントを生成して、上述の候補文字パタンを抽出することができる。

【0166】また、この発明の文字切出し方法の好適な実施例によれば、隣接するセグメント間の距離に基づいて、この距離と行高さとを比較することにより、セグメントの組を統合するかどうかを判定することができる。

【0167】また、この発明の文字切出し方法の好適な実施例によれば、ある候補文字番号から別々の全ての候補文字番号を、切出し候補位置を指定することにより辿り、その候補文字番号を配列情報としてグループ化するので、この配列情報のそれぞれを各候補文字番号列として得ることができる。そして、この候補文字番号列と文字コード格納部に記録されている文字評価値とを参照し

て、文字コード格納部に格納されている文字コードを読み出して配列させ、候補文字コード列を作成する。この候補文字コード列の作成時に、前述した文字評価値に基づいて対象とする文字コードを制限するから、探索空間の削減が図れる。

【0168】また、この発明の文字切出し方法の好適な実施例によれば、類似度を取り入れた文字評価値を用いることにより、候補文字コード列の作成時に、類似度の低い文字コードを除外することができ、従って、探索空間の削減が図れる。また、制約係数には、例えば、入力文字列に出現する文字種の文字コードを候補文字コード列の構成対象とし、入力文字列に出現しない文字種の文字コードを候補文字コード列の構成対象としない、といった情報を含ませた値を設定することにより、候補文字コード列を作成する際に対象とする文字コードの個数を削減することができ、よって、探索空間の削減が図れる。

【0169】また、この発明の文字切出し方法の好適な実施例によれば、「0」の制約係数が設定された文字コードを候補文字コード列の作成対象とせず、「1」の制約係数が設定された文字コードだけを候補文字コード列の作成対象とすることにより、探索空間の削減が図れる。

【0170】次に、この発明の文字切出し装置によれば、各候補文字の文字コードを類似度に基づいて取得する文字認識部と、取得した文字コードにその文字種に応じた文字評価値を付与する文字評価部と、その文字評価値を参照して候補文字コード列を作成する候補文字コード列作成部とを具えている。そして、上述の文字評価値を、例えば、文字認識部で取得した文字コードの文字種が、1) 入力文字列に出現する文字種であるか、2) 入力文字列に出現しない文字種であるかに基づく値とすることにより、この2)に相当する文字評価値が付与された文字コードを、候補文字コード列を作成する段階で除外することができる。よって、探索空間の削減が図れる。

【0171】また、この発明の文字切出し装置の好適な構成例によれば、候補文字パタン抽出部は、セグメントの抽出と、抽出した各セグメントの統合による新規セグメントの生成とを行うことにより、上述の候補文字パタンを抽出することができる。

【0172】また、この発明の文字切出し装置の好適な構成例によれば、セグメント統合部は、隣接するセグメント間の距離に基づいて、この距離と行高さとを比較することにより、セグメントの組を統合するかどうかを判定することができる。

【0173】また、この発明の文字切出し装置の好適な構成例によれば、上述の候補文字コード列作成部は、ある候補文字番号から別々の全ての候補文字番号を、切出し候補位置を指定することにより辿り、その候補文字番



号を配列情報としてグループ化することにより、この配列情報のそれぞれを候補文字番号列として取得することができる。また、上述の処理回路は、この候補文字番号列と文字コード格納部に記録されている文字評価値とを参照して、文字コード格納部に格納されている文字コードを読み出して配列させ、候補文字コード列を作成する。この候補文字コード列の作成時に、前述した文字評価値に基づいて対象とする文字コードを制限するから、探索空間の削減が図れる。

【0174】また、この発明の文字切出し装置の好適な構成例によれば、文字評価部は、評価値計算部により類似度を取り入れた文字評価値を求めるので、候補文字コード列の作成時に、類似度の低い文字コードを除外することができ、従って、探索空間の削減が図れる。また、制約係数には、例えば、入力文字列に出現する文字種の文字コードを候補文字コード列の構成対象とし、入力文字列に出現しない文字種の文字コードを候補文字コード列の構成対象としない、といった情報を含ませた値を設定することにより、候補文字コード列を作成する際に対象とする文字コードの個数を削減することができ、よって、探索空間の削減が図れる。

【0175】また、この発明の文字切出し装置の好適な構成例によれば、「0」の制約係数が設定された文字コードを候補文字コード列の作成対象としないで、「1」の制約係数が設定された文字コードだけを候補文字コード列の作成対象とすることができ、従って、探索空間の削減が図れる。

#### 【図面の簡単な説明】

【図1】第1の実施の形態の文字切出し装置の構成を示す図である。

【図2】実施の形態の文字切出しフローを示す図である。

【図3】入力文字列パタンの一例を示す図である。

【図4】統合前のセグメント座標テーブルを示す図である。

【図5】セグメント統合部の構成を示す図である。

【図6】実施の形態のセグメント統合処理を示す図である。

【図7】セグメント間距離の例を示す図である。

【図8】統合後のセグメント座標テーブルを示す図である。

【図9】認識結果の一例を示す図である。

【図10】各候補文字の評価結果を示す図である。

【図11】候補文字コード列作成部の構成を示す図である。

【図12】セグメントテーブルの例を示す図である。

【図13】作成関数の処理を示す図である。

【図14】候補文字番号列テーブルを示す図である。

【図15】作成される候補文字コード列の一例を示す図である。

【図16】第2の実施の形態の文字切出し装置の構成を示す図である。

【図17】第2の実施の形態の入力文字列パタンの例を示す図である。

【図18】第2の実施の形態の統合前のセグメント座標テーブルの例を示す図である。

【図19】第2の実施の形態の統合後のセグメント座標テーブルの例を示す図である。

【図20】第2の実施の形態の認識結果の一例を示す図である。

【図21】第2の実施の形態の候補文字の評価結果を示す図である。

【図22】第2の実施の形態の候補文字の評価結果を示す図である。

【図23】第2の実施の形態の候補文字の評価結果を示す図である。

【図24】第2の実施の形態のセグメントテーブルの例を示す図である。

【図25】第2の実施の形態で作成される候補文字コード列の一例を示す図である。

【図26】グラフ探索処理のフローを示す図である。

【図27】セグメントテーブルの一例を示す図である。

#### 【符号の説明】

10：画像入力部

12：候補文字パタン抽出部

14：候補文字コード列作成部

16：文字認識部

18：最良文字コード列選択部

20：座標メモリ

24：制御部

28：評価値計算部

32：文字評価部

36：画像メモリ

40：セグメント統合部

44：第2読出部

48：比較部

52：書込部

56：テーブルメモリ部

60：候補文字記録部

64：文字コード列格納部

22：文字コード格納部

26：制約係数処理部

30：評価系制御部

34：入力文字列パタン

38：セグメント抽出部

42：第1読出部

46：距離検出部

50：セグメント生成部

54：テーブル作成部

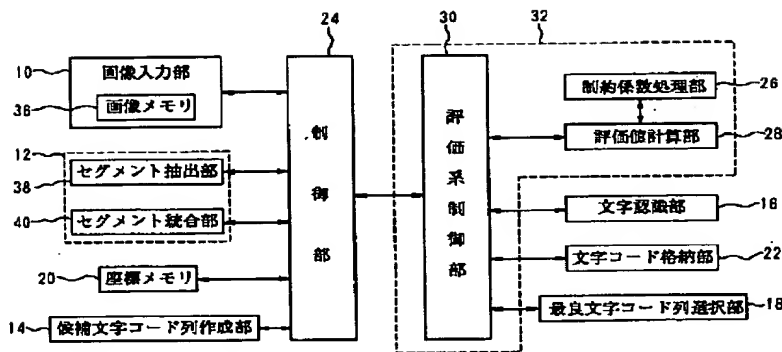
58：処理回路

62：文字番号格納部

66：単語照合部



【図1】

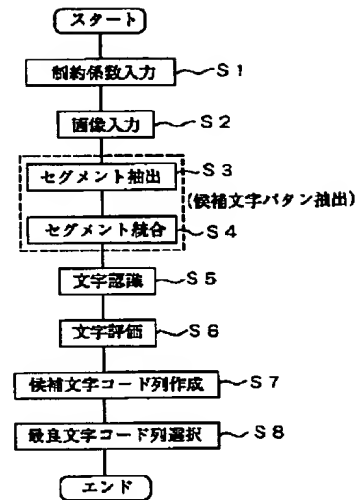


12: 候補文字ボタン抽出部

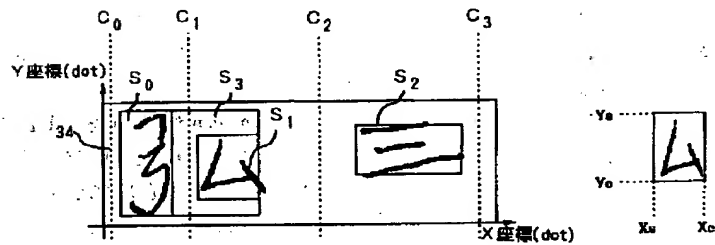
32: 文字評価部

第1の実施の形態の文字切出し装置

【図2】



【図3】



34: 入力文字列ボタン

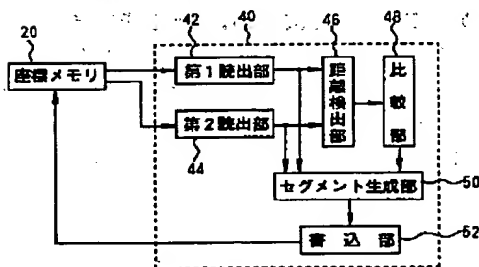
入力文字列ボタンの一例

【図4】

	Xs	Xe	Ys	Ye
S <sub>0</sub>	1	36	1	84
S <sub>1</sub>	37	106	2	89
S <sub>2</sub>	173	221	18	72

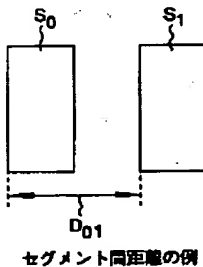
統合前のセグメント座標テーブルの例

【図5】



セグメント統合部の構成

【図7】



セグメント間距離の例

	Xs	Xe	Ys	Ye
S <sub>0</sub>	1	36	1	84
S <sub>1</sub>	37	106	2	89
S <sub>2</sub>	173	221	18	72
S <sub>3</sub>	1	106	1	89

統合後のセグメント座標テーブルの例

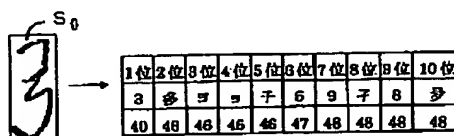
【図12】

	C <sub>0</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>
C <sub>0</sub>		S <sub>0</sub>	S <sub>3</sub>	
C <sub>1</sub>			S <sub>1</sub>	
C <sub>2</sub>				S <sub>2</sub>
C <sub>3</sub>				

※空白=NULL

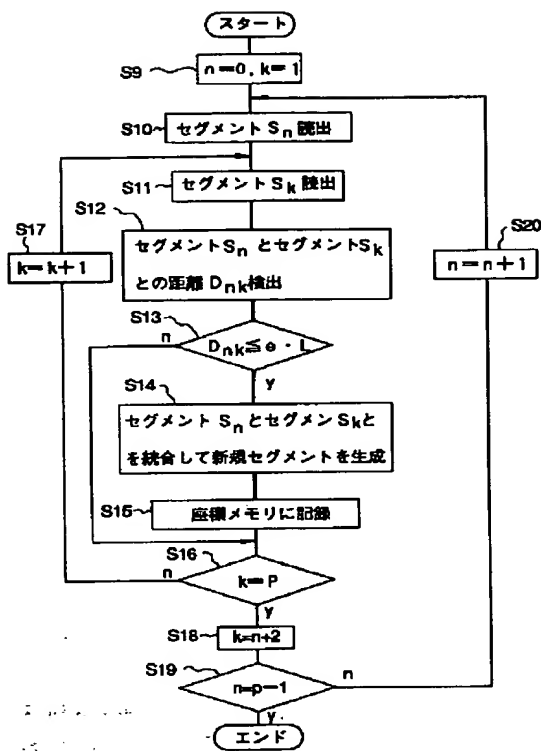
セグメントテーブルの例

【図9】



認識結果の一例

【図6】



実施の形態のセグメント統合処理

【図10】

(A)

候補文字	コード	距離	係数	評価
S <sub>0</sub>	3	40	0	NULL
	多	46	1	46
	ヨ	46	1	46
	π	46	1	46
	子	46	1	46
	5	47	0	NULL
	9	48	0	NULL
	7	48	1	48
	8	48	0	NULL
	歩	48	1	48

(B)

候補文字	コード	距離	係数	評価
S <sub>1</sub>	△	42	1	42
	く	48	1	48
	六	49	1	49
	そ	49	1	49
	点	49	1	49
	K	49	0	NULL
	ふ	60	1	60
	山	60	1	60
	人	60	1	60
	ぶ	60	1	60

(C)

候補文字	コード	距離	係数	評価
S <sub>2</sub>	三	17	1	17
	ニ	25	1	25
	ニ	29	1	29
	シ	33	1	33
	ニ	33	1	33
	ミ	40	1	40
	ラ	41	1	41
	う	42	1	42
	ら	42	1	42
	—	—	—	—

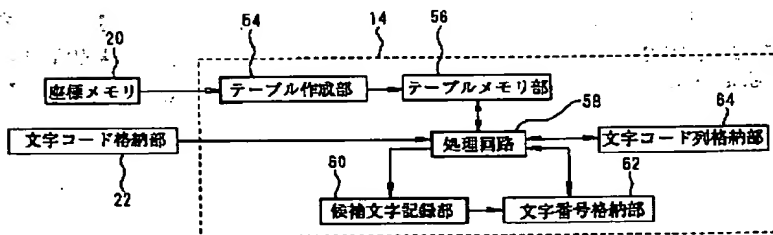
(D)

候補文字	コード	距離	係数	評価
S <sub>3</sub>	弘	32	1	32
	私	35	1	35
	松	35	1	35
	弘	38	1	38
	松	39	1	39
	支	39	1	39
	広	40	1	40
	虫	40	1	40
	法	40	1	40
	点	41	1	41

※係数 = 距離係数のこと

各候補文字の評価結果

【図11】



候補文字コード列作成部の構成

【図14】

P <sub>0</sub>	S <sub>0</sub>	S <sub>1</sub>	S <sub>2</sub>	
P <sub>1</sub>	S <sub>3</sub>	S <sub>2</sub>		

※空白=NULL

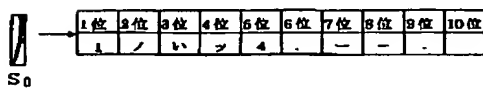
候補文字番号列テーブルの例

【図18】

	X <sub>0</sub>	X <sub>0</sub>	Y <sub>0</sub>	Y <sub>0</sub>
S <sub>0</sub>	10	28	81	105
S <sub>1</sub>	52	71	11	111
S <sub>2</sub>	128	159	56	38
S <sub>3</sub>	180	193	38	94
S <sub>4</sub>	231	241	42	67
S <sub>5</sub>	287	305	9	112

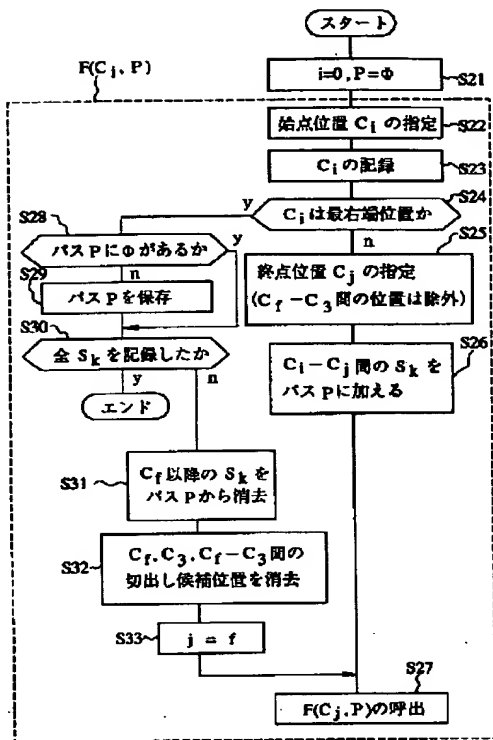
統合後のセグメント座標テーブルの例

【図20】



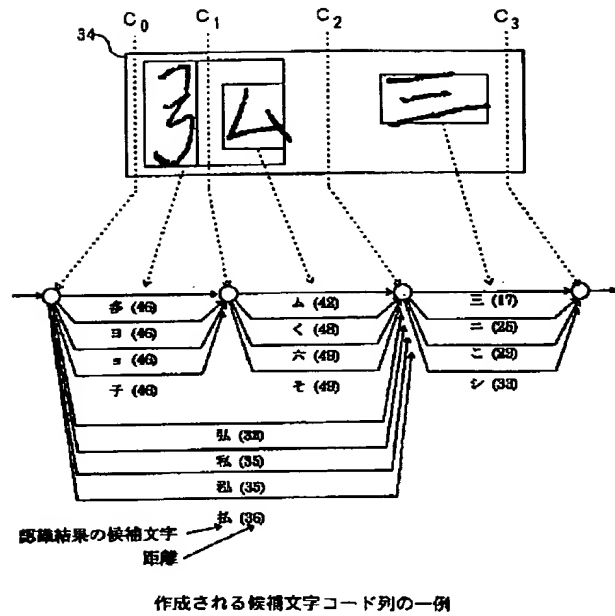
認識結果の一例

【図13】



作成関数の処理

【図15】



【図19】

	Xs	Xe	Ys	Ye
S0	10	26	61	105
S1	52	71	11	111
S2	126	159	56	58
S3	180	193	38	94
S4	231	241	42	67
S5	287	305	9	112
S6	10	71	11	111
S7	10	159	11	111
S8	52	159	11	111
S9	52	193	11	111
S10	126	193	38	94
S11	126	241	38	94
S12	180	241	38	94
S13	180	305	9	112
S14	231	305	9	112

結合後のセグメント座標テーブルの例

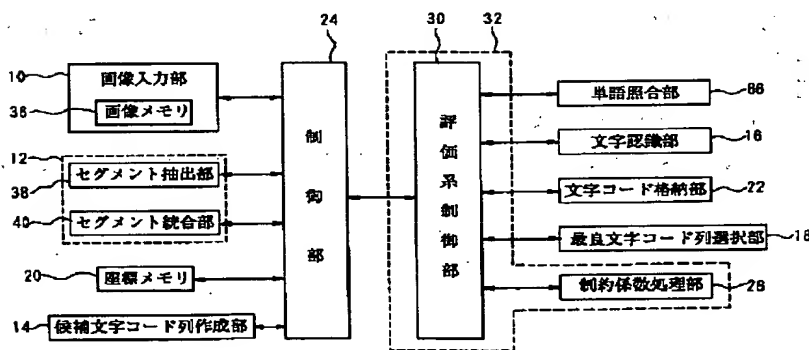
【図24】

	C0	C1	C2	C3	C4	C5	C6
C0		S0	S6	S7			
C1			S1	S8	S9		
C2				S2	S10	S11	
C3					S3	S12	S13
C4						S4	S14
C5							S5
C6							

空白= NULL

セグメントテーブルの例

【図16】



第2の実施の形態の文字切出し装置

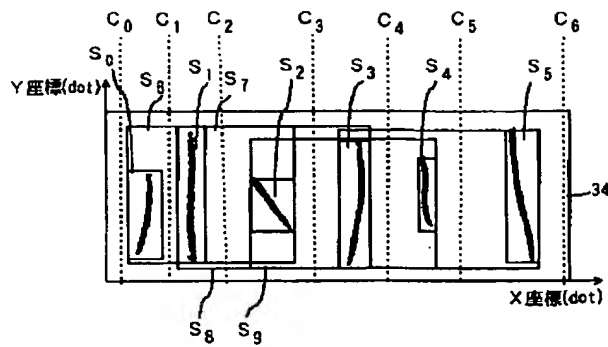
【図27】

	C1	C2	C3
C1		S1	S3
C2			S2
C3			

空白= NULL

セグメントテーブルの一例

【図17】



第2の実施の形態の入力文字列パタンの例

【図21】

(A)			(B)		
候補文字	コード	係数	候補文字	コード	係数
S0	1	0	S1	1	0
	/	0		り	0
	い	0		し	0

(C)			(D)		
候補文字	コード	係数	候補文字	コード	係数
S2	1	0	S3	1	0
	え	0		り	0
	2	0		し	0

各候補文字の評価結果（一部の候補文字略）（その1）

【図22】

(E)			(F)		
候補文字	コード	係数	候補文字	コード	係数
S4	1	0	S5	1	0
	/	0		川	1
				リ	0
				小	1
				い	0
				り	0

(G)			(H)		
候補文字	コード	係数	候補文字	コード	係数
S6	川	1	S7	小	1
	リ	0		ハ	0
	バ	0		川	1
	り	0		ハ	1
	刈	1		ふ	0
	小	1		ド	0
	イ	0		に	0
	1	0		ル	0
	州	1		州	1

各候補文字の評価結果（一部の候補文字略）（その2）

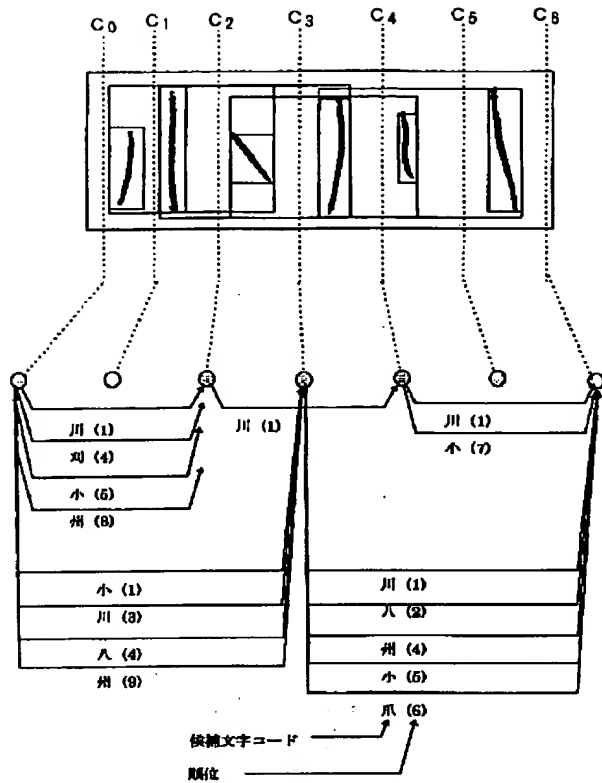
【図23】

(I)			(J)		
候補文字	コード	係数	候補文字	コード	係数
S11	い	0	S12	ハ	0
	川	1		小	1
	小	1		ハ	1
	り	0		川	1
	い	0		ド	0
	リ	0		人	1
				ふ	0
				バ	0
				リ	0
				入	1

(K)			(L)		
候補文字	コード	係数	候補文字	コード	係数
S13	川	1	S14	川	1
	ハ	0		ハ	1
	ハ	1		ハ	0
	州	1		い	0
	小	1		リ	0
	爪	1		1	0
	り	0		小	1
	い	0		り	0
	ふ	0		い	0
	ル	0		ふ	0

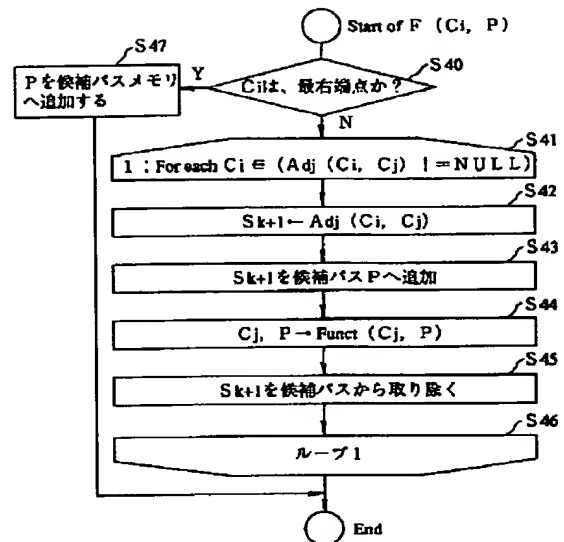
各候補文字の評価結果（一部の候補文字略）（その3）

【図25】



第2の実施の形態で作成される候補文字コード列の一例

【図26】



グラフ探索処理のフロー